APMA 0160 Lecture 9

Linear Systems of Equations

Brown University, Summer 2017

Monday, July 17

Announcements

- Exam grades online
- HW 5 due tomorrow
- HW 6 out today

Review

Solve the following systems of equations:

$$\begin{cases} 3x - y = 1 \\ -2x + 2y = 2 \end{cases}$$

$$\begin{cases} 3x - y = 1 \\ -6x + 2y = -2 \end{cases}$$

$$\begin{cases} 3x - y = 1 \\ -6x + 2y = 0 \end{cases}$$

Matrix operations

A matrix with n rows and m columns is said to have dimension $n \times m$.

Matrix operations

A matrix with n rows and m columns is said to have dimension $n \times m$.

In order to add/subtract two matrices, they must have the same dimension. Addition/subtraction is always done elementwise.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} + \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} = \begin{bmatrix} 6 & 8 \\ 10 & 12 \end{bmatrix}$$

Matrix operations

A matrix with n rows and m columns is said to have dimension $n \times m$.

In order to add/subtract two matrices, they must have the same dimension. Addition/subtraction is always done elementwise.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} + \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} = \begin{bmatrix} 6 & 8 \\ 10 & 12 \end{bmatrix}$$

Multiplication comes in two varieties: scalar and matrix. Scalar multiplication is multiplying every element of the matrix by a real number:

$$4 \begin{bmatrix} 1 & 1 \\ 4 & -2 \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} 4 & 4 \\ 16 & -8 \\ 0 & -4 \end{bmatrix}$$

Matrix multiplication

A matrix with dimension $a \times b$ can multiply a matrix with dimension $c \times d$ only if b = c. Their product will be a matrix with dimension $a \times d$. Example: product of 2×3 and 3×2 matrices produces a 2×2 matrix

$$\begin{bmatrix} 4 & -2 & 8 \\ 0 & 3 & 9 \end{bmatrix} \begin{bmatrix} 6 & -5 \\ 7 & 2 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 24 - 14 + 8 & -20 - 4 + 8 \\ 0 + 21 + 9 & 0 + 6 + 9 \end{bmatrix}$$

$$= \begin{bmatrix} 18 & -16 \\ 30 & 15 \end{bmatrix}$$

Matrix multiplication

A matrix with dimension $a \times b$ can multiply a matrix with dimension $c \times d$ only if b = c. Their product will be a matrix with dimension $a \times d$. Example: product of 2×3 and 3×2 matrices produces a 2×2 matrix

$$\begin{bmatrix} 4 & -2 & 8 \\ 0 & 3 & 9 \end{bmatrix} \begin{bmatrix} 6 & -5 \\ 7 & 2 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 24 - 14 + 8 & -20 - 4 + 8 \\ 0 + 21 + 9 & 0 + 6 + 9 \end{bmatrix}$$

$$= \begin{bmatrix} 18 & -16 \\ 30 & 15 \end{bmatrix}$$

Matrix multiplication is NOT commutative but it is associative: $AB \neq BA$ but $(AB)C = A(BC)$

Other matrix operations

Matrix transpose: $n \times m$ matrix becomes $m \times n$ (rows become columns and vice versa)

$$[4, 6, -1, 2]^T = \begin{bmatrix} 4 \\ 6 \\ -1 \\ 2 \end{bmatrix}$$

Exponents: same as multiplication n times:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix}^3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 8 & 0 \\ 0 & 0 & 27 \end{bmatrix}$$

Linear systems as matrices

All linear systems can be represented as matrices.

$$\left\{ \begin{array}{c} 3x - y = 1 \\ -2x + 2y = 2 \end{array} \right. \leftrightarrow \begin{bmatrix} 3 & -1 \\ -2 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

$$\left\{ \begin{array}{c} 12x - 4y + 8z = 0 \\ x - 7z = 12 \\ 8x + 8y - z = 9 \end{array} \right. \leftrightarrow \begin{bmatrix} 12 & -4 & 8 \\ 1 & 0 & -7 \\ 8 & 8 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0 \\ 12 \\ 9 \end{bmatrix}$$

This is nothing more than notation. Nothing special is happening.

Linear systems as matrices

All linear systems can be represented as matrices.

$$\begin{cases} 3x - y = 1 \\ -2x + 2y = 2 \end{cases} \leftrightarrow \begin{bmatrix} 3 & -1 \\ -2 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

$$\begin{cases} 12x - 4y + 8z = 0 \\ x - 7z = 12 \\ 8x + 8y - z = 9 \end{cases} \leftrightarrow \begin{bmatrix} 12 & -4 & 8 \\ 1 & 0 & -7 \\ 8 & 8 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0 \\ 12 \\ 9 \end{bmatrix}$$

This is nothing more than notation. Nothing special is happening. All linear systems can be written in the form $Ax = b$, where A is an $n \times n$ matrix, x is a $n \times 1$ vector of unknowns, and b is a $n \times 1$ vector of coefficients.

Methods for linear systems

There are many, many ways to solve linear systems. You already know several:

Methods for linear systems

There are many, many ways to solve linear systems. You already know several:

- (Gaussian) elimination/substitution
- graphing/visualizing intersections of hyperplanes
- Cramer's rule/determinants
- matrix inverse

Methods for linear systems

There are many, many ways to solve linear systems. You already know several:

- (Gaussian) elimination/substitution
- graphing/visualizing intersections of hyperplanes
- Cramer's rule/determinants
- matrix inverse

However, these are not very tractable for LARGE systems (think $n = 100$ variables). We need to look for more computationally efficient methods.

- Elimination via pivoting
- LU factorization
- iterative methods (Gauss-Seidel/Gauss-Jacobi)
- conjugate gradient

LINPACK and supercomputers

A list of the world's fastest supercomputers can be found here.

How do engineers measure their speed?

LINPACK and supercomputers

A list of the world's fastest supercomputers can be found here.

How do engineers measure their speed?

The LINPACK benchmark achieves this by asking the computer to solve a large ($n =$ hundreds of thousands or more) system of equations by pivoting and timing how long it takes.

Elimination method

We start with the observation that upper triangular systems are very simple to solve

$$\begin{bmatrix} 5 & -1 & -2 \\ 0 & 2 & 3 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 6 \\ 6 \\ 8 \end{bmatrix}$$

Elimination method

We start with the observation that upper triangular systems are very simple to solve

$$\begin{bmatrix} 5 & -1 & -2 \\ 0 & 2 & 3 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 6 \\ 6 \\ 8 \end{bmatrix}$$

Look at the bottom equation. This says that $-1z = 8$. Hence $z = -8$.

Elimination method

We start with the observation that upper triangular systems are very simple to solve

$$\begin{bmatrix} 5 & -1 & -2 \\ 0 & 2 & 3 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 6 \\ 6 \\ 8 \end{bmatrix}$$

Look at the bottom equation. This says that $-1z = 8$. Hence $z = -8$.

The middle equation is $2y + 3z = 6$. Plug in $z = -8$. Then, $y = 15$.

Elimination method

We start with the observation that upper triangular systems are very simple to solve

$$\begin{bmatrix} 5 & -1 & -2 \\ 0 & 2 & 3 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 6 \\ 6 \\ 8 \end{bmatrix}$$

Look at the bottom equation. This says that $-1z = 8$. Hence $z = -8$.

The middle equation is $2y + 3z = 6$. Plug in $z = -8$. Then, $y = 15$.

The top equation is $5x - y - 2z = 6$. Plugging in $z = -8$ and $y = 15$ gives $x = 7.4$. The system has been solved.

The process used above is called backsolving, and it can be used to solve any upper triangular (or lower triangular) matrix.

The process used above is called backsolving, and it can be used to solve any upper triangular (or lower triangular) matrix.

Since upper triangular matrices are so easy to solve, we want to learn how to transform a given system into upper triangular form. (This is also known as row echelon form or ref.)

$$\begin{bmatrix} 4 & 1 & -2 \\ 1 & 2 & 3 \\ -2 & 5 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 16 \\ 11 \\ 2 \end{bmatrix} \leftrightarrow \begin{bmatrix} 4 & 1 & -2 & 16 \\ 1 & 2 & 3 & 11 \\ -2 & 5 & 0 & 2 \end{bmatrix}$$

The matrix on the right is called an augmented matrix.

Elimination method

We need to get zeros in the first column below the first row. This can be done by multiplying the second row by 4 and subtracting.

$$\begin{bmatrix} 4 & 1 & -2 & 16 \\ 1 & 2 & 3 & 11 \\ -2 & 5 & 0 & 2 \end{bmatrix} \to \begin{bmatrix} 4 & 1 & -2 & 16 \\ 0 & -7 & -14 & -28 \\ -2 & 5 & 0 & 2 \end{bmatrix}$$

Elimination method

We need to get zeros in the first column below the first row. This can be done by multiplying the second row by 4 and subtracting.

$$\begin{bmatrix} 4 & 1 & -2 & 16 \\ 1 & 2 & 3 & 11 \\ -2 & 5 & 0 & 2 \end{bmatrix} \to \begin{bmatrix} 4 & 1 & -2 & 16 \\ 0 & -7 & -14 & -28 \\ -2 & 5 & 0 & 2 \end{bmatrix}$$

Similarly, we can get a zero in the bottom left by multiplying row 2 by 2 and adding to the first column.

$$\begin{bmatrix} 4 & 1 & -2 & 16 \\ 0 & -7 & -14 & -28 \\ -2 & 5 & 0 & 2 \end{bmatrix} \to \begin{bmatrix} 4 & 1 & -2 & 16 \\ 0 & -7 & -14 & -28 \\ 0 & 11 & -2 & 20 \end{bmatrix}$$

Elimination method

Continuing on, we need a zero in the thrid row, second column. This can be achieved by using the second row as a pivot.

$$\begin{bmatrix} 4 & 1 & -2 & 16 \\ 0 & -7 & -14 & -28 \\ 0 & 11 & -2 & 20 \end{bmatrix} \rightarrow \begin{bmatrix} 4 & 1 & -2 & 16 \\ 0 & -7 & -14 & -28 \\ 0 & 0 & \frac{-168}{11} & \frac{-168}{11} \end{bmatrix}$$

Elimination method

Continuing on, we need a zero in the thrid row, second column. This can be achieved by using the second row as a pivot.

$$\begin{bmatrix} 4 & 1 & -2 & 16 \\ 0 & -7 & -14 & -28 \\ 0 & 11 & -2 & 20 \end{bmatrix} \rightarrow \begin{bmatrix} 4 & 1 & -2 & 16 \\ 0 & -7 & -14 & -28 \\ 0 & 0 & \frac{-168}{11} & \frac{-168}{11} \end{bmatrix}$$

The system is now upper triangular. We can apply backsolving to get the solution. $z = 1$, $y = 2$, $x = 4$.

Elimination method

There is more than one way to eliminate a matrix!

$$\begin{bmatrix} 4 & 1 & -2 & 16 \\ 1 & 2 & 3 & 11 \\ -2 & 5 & 0 & 2 \end{bmatrix} \to \begin{bmatrix} 4 & 1 & -2 & 16 \\ 0 & \frac{7}{4} & \frac{7}{2} & 7 \\ -2 & 5 & 0 & 2 \end{bmatrix}$$

Elimination method

There is more than one way to eliminate a matrix!

$$\begin{bmatrix} 4 & 1 & -2 & 16 \\ 1 & 2 & 3 & 11 \\ -2 & 5 & 0 & 2 \end{bmatrix} \to \begin{bmatrix} 4 & 1 & -2 & 16 \\ 0 & \frac{7}{4} & \frac{7}{2} & 7 \\ -2 & 5 & 0 & 2 \end{bmatrix}$$

Third row:

$$\begin{bmatrix} 4 & 1 & -2 & 16 \\ 0 & \frac{7}{4} & \frac{7}{2} & 7 \\ -2 & 5 & 0 & 2 \end{bmatrix} \to \begin{bmatrix} 4 & 1 & -2 & 16 \\ 0 & \frac{7}{4} & \frac{7}{2} & 7 \\ 0 & \frac{11}{2} & -1 & 10 \end{bmatrix}$$

Elimination method

There is more than one way to eliminate a matrix!

$$\begin{bmatrix} 4 & 1 & -2 & 16 \\ 1 & 2 & 3 & 11 \\ -2 & 5 & 0 & 2 \end{bmatrix} \to \begin{bmatrix} 4 & 1 & -2 & 16 \\ 0 & \frac{7}{4} & \frac{7}{2} & 7 \\ -2 & 5 & 0 & 2 \end{bmatrix}$$

Third row:

$$\begin{bmatrix} 4 & 1 & -2 & 16 \\ 0 & \frac{7}{4} & \frac{7}{2} & 7 \\ -2 & 5 & 0 & 2 \end{bmatrix} \to \begin{bmatrix} 4 & 1 & -2 & 16 \\ 0 & \frac{7}{4} & \frac{7}{2} & 7 \\ 0 & \frac{11}{2} & -1 & 10 \end{bmatrix}$$

Third row again:

$$\begin{bmatrix} 4 & 1 & -2 & 16 \\ 0 & \frac{7}{4} & \frac{7}{2} & 7 \\ 0 & \frac{11}{2} & -1 & 10 \end{bmatrix} \to \begin{bmatrix} 4 & 1 & -2 & 16 \\ 0 & \frac{7}{4} & \frac{7}{2} & 7 \\ 0 & 0 & -12 & -12 \end{bmatrix}$$

Elimination method

The version where $R_j = R_j - mR_k$ where $k < j$ is standard elimination.

Let a_{ij} be the element of the matrix A in the ith row and jth column. We can summarize the elimination procedure as follows:

- Assume the first row has a nonzero element in its first position. Multiply each of the elements in row i in the augmented matrix by $m = a_{i1}/a_{11}$. Then, subtract from the first row so that $a_{ij} = a_{ij} - a_{1j}m$.
- Repeat this procedure for each column until the matrix is upper triangular. This creates zeros in all entries below the main diagonal.
- Starting from the bottom, use backsolving to obtain the solution of the linear system.

Pivoting

Pivoting refers to the strategic interchanging of two rows. This can prevent roundoff errors. Consider the system

$$\begin{bmatrix} 10 & -7 & 0 & 7 \\ -3 & 2.09999 & 6 & 3.90001 \\ 5 & -1 & 5 & 6 \end{bmatrix}$$

First, you can verify the true solution is $(x, y, z) = (0, -1, 1)$.

Pivoting

Let's try the elimination algorithm we just learned. We will keep 5 decimal places.

$$\begin{bmatrix} 10 & -7 & 0 & 7 \\ -3 & 2.09999 & 6 & 3.90001 \\ 5 & -1 & 5 & 6 \end{bmatrix} \rightarrow \begin{bmatrix} 10 & -7 & 0 & 7 \\ 0 & -0.00003 & 20 & 20.00003 \\ 0 & -5 & -10 & -5 \end{bmatrix}$$

Pivoting

Let's try the elimination algorithm we just learned. We will keep 5 decimal places.

$$\begin{bmatrix} 10 & -7 & 0 & 7 \\ -3 & 2.09999 & 6 & 3.90001 \\ 5 & -1 & 5 & 6 \end{bmatrix} \to \begin{bmatrix} 10 & -7 & 0 & 7 \\ 0 & -0.00003 & 20 & 20.00003 \\ 0 & -5 & -10 & -5 \end{bmatrix}$$

$$\to \begin{bmatrix} 10 & -7 & 0 & 7 \\ 0 & -0.00003 & 20 & 20.00003 \\ 0 & 0 & 20.00006 & 20 \end{bmatrix}$$

Pivoting

Let's try the elimination algorithm we just learned. We will keep 5 decimal places.

$$\begin{bmatrix} 10 & -7 & 0 & 7 \\ -3 & 2.09999 & 6 & 3.90001 \\ 5 & -1 & 5 & 6 \end{bmatrix} \to \begin{bmatrix} 10 & -7 & 0 & 7 \\ 0 & -0.00003 & 20 & 20.00003 \\ 0 & -5 & -10 & -5 \end{bmatrix}$$

$$\to \begin{bmatrix} 10 & -7 & 0 & 7 \\ 0 & -0.00003 & 20 & 20.00003 \\ 0 & 0 & 20.00006 & 20 \end{bmatrix}$$

$z = 0.99999$, $y = 7.66667$, $x = 6.06667$. What went wrong here?

Pivoting

The idea of pivoting is to rearrange rows so that the leading nonzero terms are never small.

Pivoting

The idea of pivoting is to rearrange rows so that the leading nonzero terms are never small.

Before subtracting row j , search all entries in the column for the element of greatest magnitude and swap with row j.

$$\begin{bmatrix} 10 & -7 & 0 & 7 \\ -3 & 2.09999 & 6 & 3.90001 \\ 5 & -1 & 5 & 6 \end{bmatrix} \to \begin{bmatrix} 10 & -7 & 0 & 7 \\ 0 & -0.00003 & 20 & 20.00003 \\ 0 & -5 & -10 & -5 \end{bmatrix}$$

$$\to \begin{bmatrix} 10 & -7 & 0 & 7 \\ 0 & -5 & -10 & -5 \\ 0 & -0.00003 & 20 & 20.00003 \end{bmatrix} \to \begin{bmatrix} 10 & -7 & 0 & 7 \\ 0 & -5 & -10 & -5 \\ 0 & 0 & 3.3e6 & 3.3e6 \end{bmatrix}$$

Pivoting

The idea of pivoting is to rearrange rows so that the leading nonzero terms are never small.

Before subtracting row j , search all entries in the column for the element of greatest magnitude and swap with row j.

$$\begin{bmatrix} 10 & -7 & 0 & 7 \\ -3 & 2.09999 & 6 & 3.90001 \\ 5 & -1 & 5 & 6 \end{bmatrix} \to \begin{bmatrix} 10 & -7 & 0 & 7 \\ 0 & -0.00003 & 20 & 20.00003 \\ 0 & -5 & -10 & -5 \end{bmatrix}$$

$$\to \begin{bmatrix} 10 & -7 & 0 & 7 \\ 0 & -5 & -10 & -5 \\ 0 & -0.00003 & 20 & 20.00003 \end{bmatrix} \to \begin{bmatrix} 10 & -7 & 0 & 7 \\ 0 & -5 & -10 & -5 \\ 0 & 0 & 3.3e6 & 3.3e6 \end{bmatrix}$$

Solution: $z = 1$, $y = -1$, $x = 0$. We have avoided the problem of dividing by small numbers.

Pivoting

Let a_{ij} be the element of the matrix A in the ith row and jth column. We can summarize the elimination procedure as follows:

- Assume the first row has a nonzero element in its first position. Multiply each of the elements in row i in the augmented matrix by $m = a_{i1}/a_{11}$. Then, subtract from the first row so that $a_{ij} = a_{ij} - a_{1j}m$.
- At step j, search the column for the element of greatest magnitude, then swap that row with row j.
- Repeat this procedure for each column until the matrix is upper triangular. This creates zeros in all entries below the main diagonal.
- Starting from the bottom, use backsolving to obtain the solution of the linear system.

Gauss-Jordan elimination

Gauss-Jordan elimination: keep subtracting rows until every element off of the main diagonal is zero as well. This is equivalent to finding the reduced row echelon form (rref) of the matrix.

$$\begin{bmatrix} 16 & 2 & 3 & 13 \\ 5 & 11 & 10 & 8 \\ 9 & 7 & 6 & 12 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & -3 \end{bmatrix}$$

The solution can be read off directly. However, this requires significantly more computation and is thus rarely used.

LU factorization

This technique relies on the matrix A being decomposed into a product $A = LU$, where L is lower triangular and U is upper triangular.

LU factorization

This technique relies on the matrix A being decomposed into a product $A = LU$, where L is lower triangular and U is upper triangular.

If $A = LU$, the linear system $Ax = b$ becomes $LUx = b$. Define $Ux = y$ so that $Ly = b$. Then, there are two systems of equations. However, L is lower triangular so it can be solved by forward substitution (backsolving, but in the other direction) to yield y. Then, we can backsolve $Ux = y$ to get x.

LU factorization

This technique relies on the matrix A being decomposed into a product $A = LU$, where L is lower triangular and U is upper triangular.

If $A = LU$, the linear system $Ax = b$ becomes $LUx = b$. Define $Ux = y$ so that $Ly = b$. Then, there are two systems of equations. However, L is lower triangular so it can be solved by forward substitution (backsolving, but in the other direction) to yield y. Then, we can backsolve $Ux = y$ to get x.

You might ask how to find L and U. It turns out that U is simply the reduced echelon you would normally obtain from standard elimination, while L is constructed from the multipliers used to obtain U.

LU factorization

You need to keep track of the multiplier m when subtracting m times the first row from the other rows:

$$\begin{bmatrix} 1 & -2 & 3 \\ 2 & -5 & 12 \\ 0 & 2 & 10 \end{bmatrix} \to \begin{bmatrix} 1 & -2 & 3 \\ (2) & -1 & 6 \\ 0 & 2 & 10 \end{bmatrix}$$

LU factorization

You need to keep track of the multiplier m when subtracting m times the first row from the other rows:

$$\begin{bmatrix} 1 & -2 & 3 \\ 2 & -5 & 12 \\ 0 & 2 & 10 \end{bmatrix} \to \begin{bmatrix} 1 & -2 & 3 \\ (2) & -1 & 6 \\ 0 & 2 & 10 \end{bmatrix}$$

$$\begin{bmatrix} 1 & -2 & 3 \\ (2) & -1 & 6 \\ 0 & 2 & 10 \end{bmatrix} \to \begin{bmatrix} 1 & -2 & 3 \\ (2) & -1 & 6 \\ (0) & (-2) & 2 \end{bmatrix}$$

LU factorization

Hence,

$$L = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 0 & -2 & 1 \end{bmatrix}, \; U = \begin{bmatrix} 1 & -2 & 3 \\ 0 & -1 & 6 \\ 0 & 0 & 2 \end{bmatrix}$$

You can verify $A = LU$.

Matrices in MATLAB

We have already discussed how to make matrices, but now that we're more experienced, we can learn some new commands that might save time.

Matrices in MATLAB

We have already discussed how to make matrices, but now that we're more experienced, we can learn some new commands that might save time.

`diag(vec,0)` creates a diagonal matrix, whose elements are `vec`.

$$\texttt{diag([5 -2 3],0)} \leftrightarrow \begin{bmatrix} 5 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & 3 \end{bmatrix}$$

The number 0 represents the main diagonal. You can get matrices with nonzero elements on the super/subdiagonals similarly

$$\texttt{diag([1 2],1)} \leftrightarrow \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{bmatrix}, \texttt{diag([1 2],-1)} \leftrightarrow \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 2 & 0 \end{bmatrix}$$

Matrices in MATLAB

You can add together `diag` matrices to make tridiagonal matrices

```
diag([1 1 1 1],0) + diag([2 2 2], 1) + diag([3 3 3], -1)
```

creates the tridiagonal matrix

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

Other useful commands:
`zeros(n,m)` creates a $n \times m$ matrix of zeros.

Matrices in MATLAB

Other useful commands:
`zeros(n,m)` creates a $n \times m$ matrix of zeros.

`ones(n,m)` creates a $n \times m$ matrix of ones.

Matrices in MATLAB

Other useful commands:

`zeros(n,m)` creates a $n \times m$ matrix of zeros.

`ones(n,m)` creates a $n \times m$ matrix of ones.

`eye(n)` creates the $n \times n$ identity matrix.

Matrices in MATLAB

Other useful commands:
`zeros(n,m)` creates a $n \times m$ matrix of zeros.

`ones(n,m)` creates a $n \times m$ matrix of ones.

`eye(n)` creates the $n \times n$ identity matrix.

`size(A)` will return the dimension of the matrix `A` in a vector `[rows cols]`.

Matrices in MATLAB

Sometimes, you will find it useful to write a double loop. For example, the 5×5 Hilbert matrix

$$H = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} & \frac{1}{7} \\ \frac{1}{4} & \frac{1}{5} & \frac{1}{6} & \frac{1}{7} & \frac{1}{8} \\ \frac{1}{5} & \frac{1}{6} & \frac{1}{7} & \frac{1}{8} & \frac{1}{9} \end{bmatrix}$$

can be made in a loop

MATLAB code

```
H = zeros(5,5); %initialize the matrix
for i = 1:5
    for j = 1:5
        H(i,j) = 1/(i+j-1);
    end
end
```

Linear systems in MATLAB

As we've mentioned before, linear systems in MATLAB can be solved by using the backslash operator.

MATLAB code

```
A = [3 2 -1; 2 -2 4; -1 1/2 -1];
b = [1 -2 0]'; %transpose to make a column vector
x = A\b;
```

`x` will return the solution to the system `Ax=b`. Here, `x` ends up being the column vector `[1 -2 -2]`.

Note that b must be a column vector, not a row vector. Also, the dimensions must be consistent.

Linear systems in MATLAB

The LU factorization of a matrix can be obtained by simply calling `[L,U,P] = lu(A)`. The matrix L is lower triangular, the matrix U is upper triangular, and the matrix P is a permutation matrix such that $PA = LU$ (only present if pivoting was used). Note that LU factorizations are not unique and the answer you get may differ from MATLAB's. But LU must always equal A.

Linear systems in MATLAB

The LU factorization of a matrix can be obtained by simply calling `[L,U,P] = lu(A)`. The matrix L is lower triangular, the matrix U is upper triangular, and the matrix P is a permutation matrix such that $PA = LU$ (only present if pivoting was used). Note that LU factorizations are not unique and the answer you get may differ from MATLAB's. But LU must always equal A.

MATLAB's backslash operator essentially uses LU factorization with pivoting. There are certain special types of matrices that have solution algorithms that are faster than LU factorization, though.

Tridiagonal systems

One example of such a matrix is a tridiagonal matrix.

$$\begin{bmatrix} 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -2 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -2 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -2 & 4 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2 & 4 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -2 & 4 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2 & 4 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -2 & 4 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & 4 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \end{bmatrix} = \begin{bmatrix} 12 \\ -4 \\ 8 \\ -3 \\ 0 \\ -19 \\ 2 \\ -15 \\ -27 \\ 38 \end{bmatrix}$$

Since there are many zeros and all rows look the same, there are shortcuts you can do.