Linear Algebra — basics and notation

$$A_{M\times N} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ \vdots & \vdots & & \vdots \\ a_{M1} & a_{M2} & \cdots & a_{MN} \end{bmatrix}$$

Multiplication of matrices:

$$A_{M\times N} B_{N\times P} = C_{M\times P}$$

$$(AB)_{ij} = \sum_k a_{ik} b_{kj}$$

- associative
~~addition dist~~
- distributes over addition
- not commutative, in general

Transpose: A^T $\qquad (A^T)_{ij} = a_{ji}$

$$(AB)^T = B^T A^T$$

Inverse: A^{-1} $\qquad$ if it exists, it is unique

$$(AB)^{-1} = B^{-1} A^{-1}$$

$$(A^{-1})^T = (A^T)^{-1}$$

Symmetric matrix: $\quad A = A^T$

Gaussian elimination

Take a system of equations such as:

$$\begin{aligned} 2x + y + z &= 1 \\ 4x + y &= -2 \\ -2x + 2y + z &= 7 \end{aligned} \qquad \text{or, in matrix form} \qquad \begin{bmatrix} 2 & 1 & 1 \\ 4 & 1 & 0 \\ -2 & 2 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 \\ -2 \\ 7 \end{bmatrix}$$

We can solve these equations by adding a multiple of one equation to another equation. We'll work systematically from top to bottom. To simplify things somewhat, we'll represent the equations in an "augmented matrix"

$$\left[\begin{array}{ccc|c} 2 & 1 & 1 & 1 \\ 4 & 1 & 0 & -2 \\ -2 & 2 & 1 & 7 \end{array}\right]$$

First, add a multiple of the first equation to the second and third equations to eliminate the x coefficient:

$$\left[\begin{array}{ccc|c} 2 & 1 & 1 & 1 \\ 0 & -1 & -2 & -4 \\ 0 & 3 & 2 & 8 \end{array}\right]$$

-2 times first added to second
1 times first added to third

Then, add a multiple of the second equation to the third to eliminate its y coefficient:

$$\left[\begin{array}{ccc|c} 2 & 1 & 1 & 1 \\ 0 & -1 & -2 & -4 \\ 0 & 0 & -4 & -4 \end{array}\right]$$

3 times second added to third

This is the result of <u>Gaussian elimination.</u> We can now do <u>backsubstitution</u> to solve for the variable values

$$-4z = -4 \qquad -1y - 2z = -4 \qquad 2x + y + z = 1$$

$$\text{so } z = 1 \qquad y = 4 - 2(1) \qquad x = \tfrac{1}{2} - \tfrac{1}{2}(2) - \tfrac{1}{2}(1)$$

$$\text{so } y = 2 \qquad \text{so } x = -1$$

Pivoting

When we transformed $\left[\begin{array}{ccc|c} 2 & 1 & 1 & 1 \\ 4 & 1 & 0 & -2 \\ -2 & 2 & 1 & 7 \end{array}\right]$ into $\left[\begin{array}{ccc|c} 2 & 1 & 1 & 1 \\ 0 & -1 & -2 & -4 \\ 0 & 3 & 2 & 8 \end{array}\right]$

the element "2" in the upper left corner is the <u>pivot</u>.
If we encounter a zero pivot, we cannot procede; the solution is to rearrange the equations. We can either:

- Swap two rows — this does not change the solution. This is called "partial pivoting."
- Swap rows and columns — this ~~may~~ will permute the order of the variables (because of the column swap) so we must keep track of the permutation. This is called "full pivoting"

Gaussian (and Gauss-Jordan) elimination is numerically unstable without pivoting!!!

It is not known exactly which element is the best pivot, but picking the largest (magnitude) element is known, both theoretically and practically, to be a very good choice.

The largest element, however, will depend on the original scaling of the equations. Often, numerical routines will pick the pivot to be the element that would have been the largest if the original equations had been scaled so that the largest coefficient in the equation was of unit magnitude.

One final note — partial pivoting is "almost as good" as full pivoting. (And partial pivoting is easier than full pivoting.)

Gauss–Jordan elimination

Instead of stopping with the upper triangular matrix in Gaussian elimination, we could keep going to transform it into an identity matrix and then read our solution off directly from the right side of the augmented matrix:

$$\left[\begin{array}{ccc|c} 2 & 1 & 1 & 1 \\ 0 & -1 & -2 & -4 \\ 0 & 0 & -4 & -4 \end{array}\right]$$

$$\left[\begin{array}{ccc|c} 1 & 1/2 & 1/2 & 1/2 \\ 0 & 1 & 2 & 4 \\ 0 & 0 & 1 & 1 \end{array}\right]$$

multiply first row by $1/2$
multiply second row by -1
multiply third row by $-1/4$

$$\left[\begin{array}{ccc|c} 1 & 1/2 & 0 & 0 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 1 \end{array}\right]$$

subtract 2 times third row from second
subtract $1/2$ times third row from first

$$\left[\begin{array}{ccc|c} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 1 \end{array}\right]$$

subtract $1/2$ times second row from first

This is actually not advantageous for solving a single system of linear equations because it requires more operations than Gaussian elimination with backsubstitution:

- Gaussian elimination w/ backsubstitution: $\frac{n^3}{3} + n^2 + \frac{n}{3}$
- Gauss–Jordan elimination: $\frac{n^3}{2} + \frac{n^2}{2}$

multiplication/division steps

However, Gauss–Jordan elimination is good for solving multiple systems of equations with the same left-hand sides (so long as the right hand sides are all known in advance).

Suppose we have the systems of equations

$$Ax_1 = b_1 \qquad Ax_2 = b_2 \qquad Ax_3 = b_3$$

We can then form the augmented matrix

$$[\,A \mid b_1 \mid b_2 \mid b_3\,]$$

and then do Gauss-Jordan elimination, after which the right-hand side of the augmented matrix contains the solutions for x_1, x_2, and x_3.

Calculating the matrix inverse

We can use Gauss-Jordan elimination to calculate the inverse of a matrix. Suppose we have a matrix A and want to find its inverse A^{-1}. To simplify notation, assume A (and A^{-1}) are 3×3 matrices. We decompose A^{-1} into columns:

$$A^{-1} = [\,x_1 \mid x_2 \mid x_3\,]$$

and note that computing the columns of the inverse is simply a matter of solving the three systems of equations:

$$Ax_1 = \begin{bmatrix}1\\0\\0\end{bmatrix} \qquad Ax_2 = \begin{bmatrix}0\\1\\0\end{bmatrix} \qquad Ax_3 = \begin{bmatrix}0\\0\\1\end{bmatrix}$$

We start with the augmented matrix (using the same example as before)

$$\left[\begin{array}{ccc|ccc} 2 & 1 & 1 & 1 & 0 & 0 \\ 4 & 1 & 0 & 0 & 1 & 0 \\ -2 & 2 & 1 & 0 & 0 & 1 \end{array}\right]$$

And using row operations, transform it into:

$$\left[\begin{array}{ccc|ccc} 1 & 0 & 0 & 1/8 & 1/8 & -1/8 \\ 0 & 1 & 0 & -1/2 & 1/2 & 1/2 \\ 0 & 0 & 1 & 5/4 & -3/4 & -1/4 \end{array}\right]$$

(also needed to scale the rows in this process)

Therefore:

$$A^{-1} = \frac{1}{8}\begin{bmatrix} 1 & 1 & -1 \\ -4 & 4 & 4 \\ 10 & -6 & -2 \end{bmatrix}$$

LU (and LDU) decomposition

We can formalize the row operations that we did in Gaussian elimination by representing them with matrices.

In our example, the first step was to subtract 2 times the first row from the second. This can be done by the following matrix when multiplied on the left:

$$E_{21} = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

subtract twice the first row from the second

We can verify that this works (just considering the matrix A for now)

$$E_{21}\underbrace{\begin{bmatrix} 2 & 1 & 1 \\ 4 & 1 & 0 \\ -2 & 2 & 1 \end{bmatrix}}_{A} = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 2 & 1 & 1 \\ 4 & 1 & 0 \\ -2 & 2 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 1 \\ 0 & -1 & -2 \\ -2 & 2 & 1 \end{bmatrix}$$

The other two operations can similarly be represented by matrices:

$$E_{31} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$ add the first row to the third

$$E_{32} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 3 & 1 \end{bmatrix}$$ add 3 times the second row to the third

So, our transformation of the matrix A into the upper triangular matrix U is accomplished by:

$$U = E_{32} E_{31} E_{21} A$$

We can similarly relate the transformation of U back to A by undoing each of these row operations

$$E_{32}^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -3 & 1 \end{bmatrix}$$ add -3 times the second row to the third

$$E_{31}^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}$$ subtract the first row from the third

$$E_{21}^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$ add twice the first row to the second

We must undo the operations in reverse order from the way we originally did them, so:

$$A = E_{21}^{-1} E_{31}^{-1} E_{32}^{-1} U$$

Note that all these E matrices are lower triangular because we work from the top down — always adding a multiple of a higher row to a lower row.

The product of lower triangular matrices will always be a lower triangular matrix. We can then write the relationship between A and U as

$$A = LU$$

Further note that in Gaussian elimination, we do not scale any row — only add a multiple of a higher row. Therefore, the elements on the diagonal of L will all be 1.

The elements on the diagonal of U will in general not be zero. These can be transformed to 1's by introducing a diagonal matrix that scales each row:

$$\overset{A}{\begin{bmatrix} 2 & 1 & 1 \\ 4 & 1 & 0 \\ -2 & 2 & 1 \end{bmatrix}} = \overset{L}{\begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & -3 & 1 \end{bmatrix}} \overset{U}{\begin{bmatrix} 2 & 1 & 1 \\ 0 & -1 & -2 \\ 0 & 0 & -4 \end{bmatrix}}$$

$$= \overset{L}{\begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & -3 & 1 \end{bmatrix}} \overset{D}{\begin{bmatrix} 2 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -4 \end{bmatrix}} \overset{U}{\begin{bmatrix} 1 & 1/2 & 1/2 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{bmatrix}}$$

The ~~preffer~~ preferred approach to solving a system of linear equations is to use the LU decomposition.

Given: $Ax = b$, this becomes $L(Ux) = b$

First solve for y in $Ly = b$ using forward substitution and then solve for x in $Ux = y$ using backsubstitution.

The LU decomposition of a matrix can be computed in about $\frac{1}{3}n^3$ operations — approximately the same as Gaussian elimination with backsubstitution, but the advantage is that once you have the LU decomposition, you can solve that system of equations (with different right hand sides) in $O(n^2)$ time whereas Gaussian elimination (and Gauss-Jordan elimination) would have to be run again from the start.

We can use the LU decomposition to compute the matrix inverse by solving for the columns of A^{-1} as before, i.e.

$A^{-1} = [x_1 | x_2 | \cdots | x_n]$ so solve

$$\begin{aligned} Ax_1 &= e_1 \\ Ax_2 &= e_2 \\ &\vdots \\ Ax_n &= e_n \end{aligned}$$

using the LU decomposition of A and repeated backsubstitution.

Note that if you ever need to compute the matrix $A^{-1}B$ then you should LU decompose A and backsubstitute with the columns of B (instead of $e_1 \ldots e_n$ which would compute A^{-1}). This saves a matrix multiplication ($O(n^3)$) and is more accurate.

$$A^{-1}B = [x_1 | x_2 | \cdots | x_n] \qquad \begin{aligned} B^{-1}Ax_1 &= e_1 \\ B^{-1}Ax_2 &= e_2 \\ &\vdots \end{aligned} \qquad \text{or} \qquad \begin{aligned} Ax_1 &= b_1 \\ Ax_2 &= b_2 \\ &\vdots \end{aligned}$$

Finally, note that solving a system of linear equations by first computing A^{-1} and then $A^{-1}b$ is not recommended because it is susceptible to roundoff error.

Also, the LU (and LDU) decompositions are unique.

One more thing: if A is symmetric then the LDU decomposition is $A = LDL^T$

Pivoting (again)

Pivoting is required for the stability of Crout's algorithm, which we will use to compute the LU decomposition. Partial pivoting (swapping rows) is sufficient and is all that can be implemented efficiently anyway.

To represent pivoting as a matrix, we use a permutation matrix. For example, to swap the second and third rows of a 3x3 matrix, we would use:

$$P_{23} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

For nonsingular matrices, there is always a permutation matrix P (that collects all our row swaps) so that the pivot at each step in the LU decomposition is nonzero.

$$PA = LU$$

Note that a permutation matrix has the same rows as the identity matrix.

Crout's algorithm for computing $A=LU$

We need to solve: (for the 4x4 case)

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} = \begin{bmatrix} \alpha_{11} & 0 & 0 & 0 \\ \alpha_{21} & \alpha_{22} & 0 & 0 \\ \alpha_{31} & \alpha_{32} & \alpha_{33} & 0 \\ \alpha_{41} & \alpha_{42} & \alpha_{43} & \alpha_{44} \end{bmatrix} \begin{bmatrix} \beta_{11} & \beta_{12} & \beta_{13} & \beta_{14} \\ 0 & \beta_{22} & \beta_{23} & \beta_{24} \\ 0 & 0 & \beta_{33} & \beta_{34} \\ 0 & 0 & 0 & \beta_{44} \end{bmatrix}$$

We can write each component of A as

$$a_{ij} = \alpha_{i1}\beta_{1j} + \cdots$$

After setting all the $\alpha_{ii} = 1$, there are n^2 unknowns (the β's and remaining α's) which we must solve for using n^2 equations.

Crout's algorithm calculates this solution easily just by arranging the equations in the right order. Note that the equations are:

$$\cancel{i<j \quad a_{i1} b_{1j} + a_{i2} b_{2j} + \cdots + a_{ii} b_{ij} = a_{ij}}$$

$$\cancel{i=j \quad a_{i1} + b_{1j} + a_{i2} b_{2j} + \cdots + a}$$

$$i<j \quad \alpha_{i1}\beta_{1j} + \alpha_{i2}\beta_{2j} + \cdots + \alpha_{ii}\beta_{ij} = a_{ij}$$

$$i=j \quad \alpha_{i1}\beta_{1j} + \alpha_{i2}\beta_{2j} + \cdots + \alpha_{ii}\beta_{jj} = a_{ij}$$

$$i>j \quad \alpha_{i1}\beta_{1j} + \alpha_{i2}\beta_{2j} + \cdots + \alpha_{ij}\beta_{jj} = a_{ij}$$

The algorithm is as follows:

- let $\alpha_{ii} = 1$ for $i = 1$ to n
- For $j = 1$ to n
  - For $i = 1$ to j, let $\beta_{ij} = a_{ij} - \sum_{k=1}^{i-1} \alpha_{ik}\beta_{kj}$
  - For $i = j+1$ to n, let $\alpha_{ij} = \frac{1}{\beta_{jj}}\left(a_{ij} - \sum_{k=1}^{j-1} \alpha_{ik}\beta_{kj}\right)$

Note that all the α's and β's needed on the right hand side of these equations are already determined before they are needed.

Rank and singular matrices

So far, we have assumed that we can always have a nonzero pivot (which we might have to get by pivoting). If this is so (and we only need to try partial pivoting to do this), then we consider the matrix <u>nonsingular</u>. This also implies that there is a ~~q~~ unique solution to the system of equations.

However, if partial pivoting cannot get us a nonzero pivot, then A is singular, and the system of equations is underdetermined.

A nonsingular matrix is always invertible (as we have demonstrated with Gauss-Jordan elimination). Furthermore every invertible matrix is nonsingular.

For a singular matrix, we can still use row operations to put it into a sort of upper triangular form called <u>echelon form</u>. When we cannot find a nonzero pivot in one column, we simply go to the next column.

For example,

$$\begin{bmatrix} 1 & 3 & 1 & 5 \\ 1 & 3 & 3 & 9 \\ 2 & 6 & 0 & 6 \\ -1 & -3 & 1 & -1 \end{bmatrix} \text{ can be transformed into...} \begin{bmatrix} 1 & 3 & 1 & 5 \\ 0 & 0 & 2 & 4 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

There will be at least one row of zeros in the echelon form of a singular matrix. The number of nonzero rows is the <u>rank</u> of the matrix.

We can now extend the LU decomposition to nonsquare matrices. For a matrix $A_{M \times N}$, there exist matrices P, L, U such that:

$$P_{M \times M} A_{M \times N} = L_{M \times M} U_{M \times N}$$

Consider now a system of equations with more variables than equations ($N > M$), for example:

$$\begin{bmatrix} 1 & 3 & 3 & 2 \\ 2 & 6 & 9 & 5 \\ -1 & -3 & 3 & 0 \end{bmatrix} \begin{bmatrix} u \\ v \\ w \\ y \end{bmatrix} = \begin{bmatrix} 1 \\ 5 \\ 5 \end{bmatrix}$$

Forming an augmented matrix and doing row operations to reach echelon form results in

$$\left[\begin{array}{cccc|c} 1 & 3 & 3 & 2 & 1 \\ 0 & 0 & 3 & 1 & 3 \\ 0 & 0 & 0 & 0 & 0 \end{array}\right]$$

Writing these equations explicitly gives us:

$$3w + y = 3 \quad \text{or} \quad w = 1 - \tfrac{1}{3}y$$

$$u + 3v + 3w + 2y = 1 \quad \text{or} \quad u = -2 - 3v - y$$

which we can write as

$$x = \begin{bmatrix} u \\ v \\ w \\ y \end{bmatrix} = \begin{bmatrix} -2 \\ 0 \\ 1 \\ 0 \end{bmatrix} + v \begin{bmatrix} -3 \\ 1 \\ 0 \\ 0 \end{bmatrix} + y \begin{bmatrix} -1 \\ 0 \\ -1/3 \\ 1 \end{bmatrix}$$

Note that the "free variables" v and y correspond to columns with no nonzero pivot.

This solution is a combination of:

- a particular solution to $Ax = b$ (the vector $(-2, 0, 1, 0)$)
- a homogeneous solution to $Ax = 0$ (the rest)

Indeed, if we consider the system $Ax=0$, reduction to echelon form is the same as before, and we get:

$$\left[\begin{array}{cccc|c} 1 & 3 & 3 & 2 & 0 \\ 0 & 0 & 3 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{array}\right]$$

Now our equations are:

$$3w + y = 0 \quad \text{or} \quad w = -\tfrac{1}{3}y$$

$$u + 3v + 3w + 2y = 0 \quad \text{or} \quad u = -3v - y$$

which gives us:

$$x = \begin{bmatrix} u \\ v \\ w \\ y \end{bmatrix} = v \begin{bmatrix} -3 \\ 1 \\ 0 \\ 0 \end{bmatrix} + y \begin{bmatrix} -1 \\ 0 \\ -1/3 \\ 1 \end{bmatrix}$$

References

Gilbert Strang, "Linear Algebra and its Applications," second edition, Harcourt Brace Jovanovich, 1980. Sections 1.1 – 1.5, 2.2

William H. Press et al., "Numerical Recipes in C," second edition, Cambridge University Press, 1992. Sections 2.1 – 2.3