LINEAR SYSTEMS AND MATRIX ALGEBRA

MATH 196, SECTION 57 (VIPUL NAIK)

Corresponding material in the book: Section 1.3.

Executive summary

(1) The *rank* of a matrix is defined as the number of nonzero rows in its reduced row-echelon form, and is also equal to the number of leading variables. The rank of a matrix is less than or equal to the number of rows. It is also less than or equal to the number of columns.
(2) The rank of the coefficient matrix of a system of simultaneous linear equations describes the number of independent equational constraints in the system.
(3) How far the coefficient matrix is from having full column rank determines the dimension of the solution space if it exists.
(4) How far the coefficient matrix is from having full row rank determines the probability that the system is consistent, roughly speaking. If the coefficient matrix has full row rank, then the system is consistent for all outputs. Otherwise, it is consistent only for some outputs and inconsistent for others.
(5) For a consistent system, the dimension of the solution space equals (number of variables) - (rank).
(6) There is a concept of "expected dimension" which is (number of variables) - (number of equations). Note that if the system does not have full row rank, the expected dimension is less than the actual dimension (if consistent). The expected dimension can be thought of as averaging the actual dimensions over all cases, where inconsistent cases are assigned dimensions of $-\infty$. This is hard to formally develop, so we will leave this out.
(7) There are various terms commonly associated with matrices: zero matrix, square matrix, diagonal, diagonal matrix, scalar matrix, identity matrix, upper-triangular matrix, and lower-triangular matrix.
(8) A vector can be represented as a row vector or a column vector.
(9) We can define a dot product of two vectors and think of it in terms of a sliding snake.
(10) We can define a matrix-vector product: a product of a matrix with a column vector. The product is a column vector whose entries are dot products of the respective rows of the matrix (considered as vectors) with the column vector.
(11) Matrix-vector multiplication is linear in the vector.
(12) A linear system of equations can be expressed as saying that the coefficient matrix times the input vector column (this is the column of unknowns) equals the output vector column (this is the column that would be the last column in the augmented matrix).

1. Rank of a matrix

The rank of the coefficient matrix is an important invariant associated with a system of linear equations.

1.1. **Rank defined using reduced row-echelon form.** If a matrix is already in reduced row-echelon form, its rank is defined as the number of leading variables, or equivalently, as the number of rows that are not all zero.

For an arbitrary matrix, we first convert it to reduced row-echelon form, then measure the rank there. Although we didn't establish it earlier, the reduced row-echelon form is unique, hence the rank defined this way makes sense and is also unique.

Note that when we used row reduction as a process to solve systems of linear equations, we had a coefficient matrix as well as an augmented matrix. The operations were determined by the coefficient matrix, but we performed them on the augmented matrix. For our discussion of rank here, we are concerned *only* with the coefficient matrix part, not with the augmenting column.

Note that:

Rank of a matrix $\leq$ Total number of rows in the matrix

Also:

Rank of a matrix $\leq$ Total number of columns in the matrix

Combining these, we obtain that:

Rank of a matrix $\leq$ Minimum of the total number of rows and the total number of columns in the matrix

Conceptually, the rank of the coefficient matrix signifies the number of independent equational constraints in there. The above two statements are interpreted as saying:

Number of independent equational constraints $\leq$ Total number of equational constraints

Number of independent equational constraints $\leq$ Total number of variables

The former makes obvious sense, and the latter makes sense because (in case the system is consistent) the dimension of the solution space is the difference (Total number of variables) - (Number of independent equational constraints).

Combining these, we obtain that:

Number of independent equational constraints $\leq$ Minimum of the number of variables and number of equations

A couple more terms:

- A matrix is said to have *full row rank* if its rank equals its number of rows. If the coefficient matrix of a system of simultaneous linear equations has full row rank, this is equivalent to saying that all the equations are independent, i.e., the system of equations has no redundancies and no inconsistencies.
- A matrix is said to have *full column rank* if its rank equals its number of columns. If the coefficient matrix of a system of simultaneous equations has full column rank, this is equivalent to saying that there are as many constraints as variables. Conceptually, this means that, *if consistent*, the system is precisely determined, and there is a unique solution. (Note: one could still have inconsistent due to zero rows in the coefficient matrix with a nonzero entry in the augmenting column; full column rank does not preclude this).
- A square matrix (i.e., a matrix with as many rows as columns) is said to have *full rank* if its rank equals its number of rows and also equals its number of columns.

1.2. **Playing around with the meaning of rank.** Converting a matrix to reduced row-echelon form (using Gauss-Jordan elimination) in order to be able to compute its rank seems tedious, but it is roughly the quickest way of finding the rank in general. If the goal is just to find the rank, some parts of the process can be skipped or simplified. For instance, we can use Gaussian elimination to get the row-echelon form instead of using Gauss-Jordan elimination to get the *reduced* row-echelon form. There do exist other algorithms that are better suited to finding the rank of a matrix, but now is not the time to discuss them.

The only way a matrix can have rank zero is if all the entries of the matrix are zero. The case of matrices of rank one is more interesting. Clearly, if there is only one nonzero row in the matrix, then it has rank one. Another way a matrix can have rank one is if all the nonzero rows in the matrix are scalar multiples of each other. Consider, for instance, the matrix:

$$\begin{bmatrix} 1 & 2 & 5 \\ 3 & 6 & 15 \end{bmatrix}$$

Note that each entry of the second row is three times the corresponding column entry in the first row. If the coefficient matrix of a linear system looks like this, the coefficient part of the second equation is a carbon copy of the first, with a tripling smudge. If we think in terms of row reduction, then subtracting 3 times the first equation from the second equation gives us:

$$\begin{bmatrix} 1 & 2 & 5 \\ 0 & 0 & 0 \end{bmatrix}$$

Note now that the second row is completely zero. This is because, when we got rid of the multiple of the first row, nothing survived.

This means that if we have a system of equations:

$$\begin{aligned} x + 2y + 5z &= a \\ 3x + 6y + 15z &= b \end{aligned}$$

then the system will either be redundant (this happens if $b = 3a$) or inconsistent (this happens if $b \neq 3a$). In the former case, since we effectively have only one constraint, the solution set is two-dimensional (using $3 - 1 = 2$). In the latter case, the solution set is empty.

1.3. **Row rank and its role in determining consistency.** If the coefficient matrix has *full row rank*, then the system of linear equations is consistent. This can be seen by looking at the reduced row-echelon form.

On the other hand, if the coefficient matrix does not have full row rank, this means that applying Gauss-Jordan elimination makes one or more of its rows zero. For each of the zero rows in the coefficient matrix, consider the corresponding output value (the augmented matrix entry). If that is zero, all is well, and the constraint is essentially a no-information constraint. It can be dropped. If, on the other hand, the output value is nonzero, then that equation gives a contradiction, which means that the system as a whole is inconsistent, i.e., it has no solutions.

The upshot is that, if the coefficient matrix does not have full row rank, there are two possibilities:

- After doing Gauss-Jordan elimination, the augmented matrix entries for all the zero rows in the coefficient matrix are zero. In this case, the system is consistent. We can discard all the zero rows and solve the system comprised of the remaining rows. The dimension of the solution space equals the number of non-leading variables, as usual.
- After doing Gauss-Jordan elimination, the augmented matrix entries for at least one of the zero rows is nonzero. In this case, the system is inconsistent, or equivalently, the solution space is empty.

This means that if the coefficient matrix does not have full row rank, then the consistency of the system hinges on the last column of the augmented matrix.

1.4. **Column rank and its role in determining solution space dimensions.** The relation between the rank and the number of columns affects the dimension of the solution space assuming the system is consistent. As noted above, if the system is consistent, we can discard the zero rows and get that the dimension of the solution space equals the number of non-leading variables. In other words:

Dimension of solution space = Number of non-leading variables = (Number of variables) - (Rank) = (Number of columns) - (Rank)

1.5. **Summary.** The following is a useful summary of the state-of-the-art in terms of our understanding of how the coefficient matrix and augmented matrix control the nature of the set of solutions to a linear system:

(1) If the coefficient matrix of a linear system has full row rank, then the system is consistent regardless of the output column. The dimension of the solution space is given as (number of variables) - (number of equations).
(2) If the coefficient matrix of a linear system has full column rank, then the system is either inconsistent or has a unique solution.
(3) If the coefficient matrix of a linear system with an equal number of variables and equations has full row rank and full column rank, then the system has a unique solution.
(4) If the coefficient matrix of a linear system does not have full row rank, then the system may or may not be consistent. Whether it is consistent or not depends on the nature of the output column. If the output column is such that after converting to rref all the zero rows give zero outputs, the system is consistent. Otherwise, it is inconsistent. Conditional to the system being consistent, the dimension of the solution space is (number of variables) - (rank).
(5) We can capture these by a concept of "expected dimension" of a system. Define the expected dimension of a system as (number of variables) - (number of equations). In case of full row rank, the expected dimension equals the actual dimension. In case the row rank is not full, the actual dimension is not the same as the expected dimension. Either the solution set is empty or it has

dimension (number of variables) - (rank), which is higher than the expected dimension. The expected dimension can be thought of as a kind of "averaging out" of the actual dimensions over all possible output columns, where we assign an empty solution set a dimension $-\infty$.

2. Matrices and vectors: some terminology

2.1. **The purpose of the terminology.** We know that a large part of the behavior of a linear system is controlled by its coefficient matrix. Therefore, we should ideally be able to look at the coefficient matrix, which *prima facie* is just a collection of numbers, and understand the story behind those numbers. In order to do that, it helps to develop some terminology related to matrices.

2.2. **Matrices and their entries.** A $m \times n$ matrix is a matrix with m rows and n columns. Typically, we use subscripts for the matrix entries with the first subscript denoting the row number and the second subscript denoting the column number. For instance, a matrix $A = (a_{ij})$ would mean that the entry a_{ij} describes the entry in the i^{th} row and j^{th} column. Explicitly, the numbering for a 2×3 matrix is as follows:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix}$$

Note that a $m \times n$ matrix and a $n \times m$ matrix look different from each other. The roles of rows and columns, though symmetric in a cosmic sense, are *not* symmetric at the level of nitty-gritty computation. For many of the ideas that we will discuss in the near and far future, it is *very important to get clear in one's head whether we're talking, at any given instant, about the number of rows or the number of columns.*

2.3. **Vectors as row matrices, column matrices, and diagonal matrices.** A vector is a tuple of numbers (with only one dimension of arrangement, unlike matrices, that use two dimensions to arrange the numbers). A vector could be represented as a matrix in either of two standard ways. It could be represented as a *row* vector, where all the numbers are written in a row:

$$\begin{bmatrix} a_1 & a_2 & \dots & a_n \end{bmatrix}$$

Note that a vector with n coordinates becomes a $1 \times n$ matrix when written as a row vector.

It could also be written as a *column* vector:

$$\begin{bmatrix} a_1 \\ a_2 \\ \cdot \\ \cdot \\ \cdot \\ a_n \end{bmatrix}$$

Note that a vector with n coordinates becomes a $n \times 1$ matrix when written as a column vector.

There is a third way of writing vectors as matrices, which may strike you as a bit unusual right now, but is particularly useful for certain purposes. This is to write the vector as a *diagonal* matrix. Explicitly, a vector with coordinates $a_1, a_2, \dots, a_n$ is written as the matrix:

$$\begin{bmatrix} a_1 & 0 & 0 & \dots & 0 \\ 0 & a_2 & 0 & \dots & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \dots & 0 & a_n \end{bmatrix}$$

See the below section for the definition of diagonal matrix.

2.4. **Square matrices, diagonal matrices, and upper/lower-triangular matrices.** A matrix is termed a *square* matrix if it has an equal number of rows and columns.

The following is some terminology used in the context of square matrices.

- A matrix is termed the *zero* matrix if all its entries are zero.
- The *diagonal* (also called the main diagonal or principal diagonal) of a square matrix is the set of positions (and entries) where the row number and column number are equal. The diagonal includes the top left and the bottom right entry. In symbols, the diagonal entries are the entries a_{ii}.
- A square matrix is termed a *diagonal* matrix if all the entries that are *not* on the diagonal are zero. The entries on the diagonal may be zero or nonzero. In symbols, a matrix (a_{ij}) is diagonal if $a_{ij} = 0$ whenever $i \neq j$.
- A square matrix is termed the *identity* matrix if it is a diagonal matrix and all diagonal entries are equal to 1.
- A square matrix is termed a *scalar* matrix if it is a diagonal matrix and all the diagonal entries are equal to each other.
- A square matrix is termed an *upper triangular matrix* if all the entries "below" the diagonal, i.e., entries of the form $a_{ij}, i > j$, are equal to zero. It is termed a *strictly upper triangular matrix* if the diagonal entries are also zero.
- A square matrix is termed a *lower triangular matrix* if all the entries "above" the diagonal, i.e., entries of the form $a_{ij}, i < j$, are equal to zero. It is termed a *strictly lower triangular matrix* if the diagonal entries are also zero.

Note that a diagonal matrix can be defined as a matrix that is both upper and lower triangular.

2.5. **The matrix and the equational system: diagonal and triangular matrices.** A while back, when describing equation-solving in general, we had discussed *diagonal systems* (where each equation involves only one variable, and the equations involve different variables). We also described *triangular systems* (where there is one equation involving only one variable, another equation involving that and another variable, and so on).

We can now verify that:

- A system of simultaneous linear equations is a diagonal system if and only if, when the equations and variables are arranged in an appropriate order, the coefficient matrix is a diagonal matrix.
- A system of simultaneous linear equations is a triangular system if and only if, when the equations and variables are arranged in an aprpopriate order, the coefficient matrix is an upper triangular matrix, and if and only if, for another appropriate order, the coefficient matrix is a lower triangular matrix. The *upper triangular* case means that we have to solve for the last variable first (using the last equation), then solve for the second last variable (using the second last equation) and so on. The *lower triangular* case means that we have to solve for the first variable first (using the first equation), then solve for the second variable (using the second equation), and so on.

  Note that the description we gave for Gaussian elimination is similar to the upper triangular description, in the sense that if the system has full row rank and full column rank, then Gaussian elimination (*not* the full Gauss-Jordan elimination) will yield an upper triangular matrix.

  The upper triangular and lower triangular cases are conceptually equivalent. However, to maintain clarity and consistency with the way we formulated Gaussian elimination, we will describe future results in the language of the upper triangular case.

2.6. **Facts about ranks.** The following facts about matrices are useful to know and easy to verify:

- Any diagonal matrix with all diagonal entries nonzero has full rank. More generally, for a diagonal matrix, the rank is the number of nonzero diagonal entries.
- Any nonzero scalar matrix has full rank.
- Any upper triangular or lower triangular matrix where all the diagonal entries are nonzero has full rank. More generally, for an upper triangular or lower triangular matrix, the rank is the number of nonzero diagonal entries.

3. Dot product, matrix-vector product, and matrix multiplication

3.1. **Structure follows function: creating algebraic rules that capture our most common manipulation modes.** One of our main goals is understanding functional forms that are linear (which may mean linear in the variables or linear in the parameters, depending on what our purpose is), and an intermediate and related goal is solving systems of simultaneous linear equations that arise when trying to determine the inputs or the parameters. As we have seen, the structure of such systems is controlled by the nature of the coefficient matrix. There are typical modes of manipulation associated with understanding these systems of equations.

We want to build an algebraic structure on the collection of matrices such that the operations for that structure reflect the common types of manipulation we need to do with linear systems. Once we have defined the operations, we can study the abstract algebraic properties of these operations, and use that information to derive insight about what we're ultimately interested in: modeling and equations.

If you do *not* keep the connection closely in mind, the matrix operations feel very arbitrary, like *ad hoc* number rules. And these rules won't help you much because you won't understand *when* to do *what* operation. With that said, keep in mind that sometimes we will not be able to explain the connection immediately. So there may be a brief period in time where you really have to just store it as a formal definition, before we explore what it really means. But make sure it's a brief period, and that you do understand what the operations means.

3.2. **The dot product as linear in both sets of variables separately.** Consider variables $a_1, a_2, \dots, a_m$ and separately consider variables $b_1, b_2, \dots, b_m$. The dot product of the vectors $\langle a_1, a_2, \dots, a_m \rangle$ and $\langle b_1, b_2, \dots, b_m \rangle$ is denoted and defined as follows:

$$\langle a_1, a_2, \dots, a_m \rangle \cdot \langle b_1, b_2, \dots, b_m \rangle = \sum_{i=1}^{m} a_i b_i$$

The expression for the dot product is jointly linear in all the a_is. It is also jointly linear in all the b_is. It is, however, not linear in the a_is and b_is together (recall the earlier distinction between affine linear and affine multilinear) because each a_i interacts with the corresponding b_i through multiplication.

The dot product can be used to model the relationship between the variables and the coefficients in a homogeneous linear function (here, "homogeneous" means the intercept is zero) and more generally in the homogeneous part of a linear function.

3.3. **Row and column notation for vectors and thinking of the dot product as a sliding snake.** We can think of the dot product of two vectors as follows:

$$\begin{bmatrix} a_1 & a_2 & \dots & a_m \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ \cdot \\ \cdot \\ \cdot \\ b_m \end{bmatrix}$$

We can think of a "sliding snake" that moves along the row for the entity being multiplied on the left and moves along the column for the entity being multiplied on the right.

We can now use this to define matrix-vector multiplication of a $n \times m$ matrix with a $m \times 1$ (i.e., a column matrix with m entries). Denote by a_{ij} the entry :

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ \cdot \\ \cdot \\ \cdot \\ b_m \end{bmatrix}$$

The product is now no longer a number. Rather, it is a column matrix, given as:

$$\begin{bmatrix} \sum_{i=1}^{m} a_{1i}b_i \\ \sum_{i=1}^{m} a_{2i}b_i \\ \cdot \\ \cdot \\ \cdot \\ \sum_{i=1}^{m} a_{ni}b_i \end{bmatrix}$$

Basically, for each entry of the product (a column vector) we multiply the corresponding row in the matrix with the column vector.

NOTE: VERY IMPORTANT: The vector being multiplied is a $m \times 1$ column vector and the result of the multiplication is a $n \times 1$ column vector. Note the potential difference in the number of coordinates between the vector of b_is and the output vector. It's very important to have a sense for these numbers. Eventually, you'll get it. But the earlier the better. It's not so important which letter is m and which is n (the roles of the letters could be interchanged) but the relative significance within the context is important. Also, note that the $n \times m$ matrix starts with a $m \times 1$ input and gives a $n \times 1$ output. In other words, the matrix dimensions are (output dimension) X (input dimension). The number of rows is the output dimension, and the number of columns is the input dimension. Make sure you understand this *very clearly* at the procedural level.

3.4. **Matrix multiplication: a teaser preview.** Let A and B be matrices. Denote by a_{ij} the entry in the i^{th} row and j^{th} column of A. Denote by b_{jk} the entry in the j^{th} row and k^{th} column of B.

Suppose the number of columns in A equals the number of rows in B. In other words, suppose A is a $m \times n$ matrix and B is a $n \times p$ matrix. Then, AB is a $m \times p$ matrix, and if we denote it by C with entries c_{ik}, we have the formula:

$$c_{ik} = \sum_{j=1}^{n} a_{ij} b_{jk}$$

The sliding snake visualization of the rule continues to work. For the ik^{th} entry of the product matrix C that we need to fill, we have a sliding snake that is moving horizontally along the i^{th} row of A and simultaneously moving vertically along the k^{th} column of B.

The previous cases of dot product (vector-vector multiplication) and matrix-vector multiplication can be viewed as special cases of matrix-matrix multiplication.

We will return to a more detailed treatment of matrix multiplication a week or so from now.

3.5. **Linearity of multiplication.** Matrix-vector multiplication is linear in terms of both the matrix and the vector (this is part of a more general observation that matrix multiplication is linear in terms of both the matrices involved, but that'll have to wait for later). Explicitly, the following are true for a $n \times m$ matrix A:

- For any m-dimensional vectors $\vec{x}$ and $\vec{y}$ (interpreted as column vectors), we have $A(\vec{x}+\vec{y}) = A\vec{x}+A\vec{y}$. The addition of vectors is done coordinate-wise. Note that the addition of $\vec{x}$ and $\vec{y}$ is addition of vectors both with m coordinates. The addition of $A\vec{x}$ and $A\vec{y}$ is addition of vectors with n coordinates.
- For any m-dimensional vector $\vec{x}$ (interpreted as a column vector) and any scalar λ, we have $A(\lambda\vec{x}) = \lambda(A\vec{x})$. Here, the scalar-vector product is defined as multiplying the scalar separately by each coordinate of the vector.

Again, keep in mind that the roles of m and n could be interchanged in this setting. But if you're interchanging them, *interchange them throughout*. The letters aren't sacred, but the relative roles of input and output are.

3.6. **Linear system as a statement about a matrix-vector product.** Any system of linear equations can be expressed as:

(Coefficient matrix)(Vector of unknowns) = (Vector of outputs (this would be the last column in the augmented matrix))

If the vector of unknowns is denoted $\vec{x}$, the coefficient matrix is denoted A, and the vector of outputs is denoted $\vec{b}$, we write this as:

$$A\vec{x} = \vec{b}$$

Here, both A and $\vec{b}$ are known, and we need to find $\vec{x}$.

What we would ideally like to do is write:

$$\vec{x} = \frac{\vec{b}}{A}$$

Does this make sense? Not directly. But it makes approximate sense. Stay tuned for more on this next week.