

Some Revised Notes on Linear Algebra

Leon van Dommelen

October 15, 2014

Abstract

This are some revised notes on linear algebra. They supplement the current text book [2].

Contents

1	Gaussian Elimination	2
2	Partial Pivoting	2
3	Back substitution	3
4	LU Theorem	3
5	Reduction to Row Canonical form	5
6	Finding null spaces and solution spaces	5
7	Finding inverses using GE	7
8	Finding inverses using minors	7
9	Finding inverses using transposing	8
10	Finding eigenvalues	8
11	Eigenvectors of nonsymmetric matrices	9
12	Eigenvectors of symmetric matrices	9
13	Change of basis (coordinate system)	11

14	Diagonalization of nonsymmetric matrices	14
15	Diagonalization of symmetric matrices	14

1 Gaussian Elimination

To do Gaussian (or forward) elimination in this class, the general procedure consists of four basic steps. Take the initial “submatrix” to look at to be the complete given matrix. Then:

GE I If there is only one row left in the submatrix currently being looked at, you are done with Gaussian elimination. And so you are if there are no more nonzero coefficients in the submatrix.

GE II In the submatrix currently being looked at, identify the *first* column that still has a nonzero coefficient. Any earlier columns no longer appear anywhere, and should be dropped from the submatrix being looked at.

GE III In the submatrix currently being looked at, use the *first* row to eliminate the *first* coefficient from the *subsequent* rows. Normally, you do that by subtracting suitable multiples of the first row from the subsequent ones. Step GE III can be described as “creating zeros below the pivot.” Note that this is only possible if the coefficient in the first column of the first row, called the *pivot*, is nonzero. In general, before doing the actual step GE III, you may need to exchange the current *first* row with a *later* one that will produce a better pivot. The next section will explain that in more detail.

GE IV Drop the first row and the first column from the submatrix being looked at. With the so-reduced submatrix, repeat the process starting from step GE I.

This process will give rise to a so-called “echelon form” of the matrix. That is a special case of an upper triangular (if square) or upper trapezoidal (if not square) matrix.

Additional note: the eliminations in GE II can be done in a single step for all rows below the pivot. You do not have to show each row using a separate matrix. You must however explicitly show the multiplier(s) used.

2 Partial Pivoting

Partial pivoting is defined as interchanging the *first* row of the submatrix currently being looked at with a subsequent one. Its purpose is to replace the

current pivot by a more desirable one. In particular, partial pivoting *must* be used if otherwise the pivot would be zero.

More generally, partial pivoting must be used to select the equation with the most desirable pivot. In particular

- In a *numerical* solution of systems of equations on a computer, the main objective is to minimize the effect of round-off errors. Here you should select the row having the pivot with the largest *absolute* value. (This assumes that the equations are scaled in a consistent, reasonable way.)
- In hand computations with integer matrices in homework or an exam in this class, the objective is to avoid fractions in the algebra. So you try to select a pivot which is ± 1 .

Note that there may not be such a pivot. In that case, you may be still be able to avoid fractions by partial pivoting. Or else you might be able to avoid them by multiplying the row being replaced by an integer greater than 1 instead of 1.

If that works, it should *normally* be done. However, if you want to identify matrices L and U in the LU theorem, you can only multiply the row being replaced by 1. Otherwise the theorem does not work. So you may have to live with fractions.

3 Back substitution

To solve an echelon system resulting from Gaussian elimination, start with the last equation and work backwards to the first. At any stage,

- If all the coefficients of the unknowns are zero, and the right hand side too, the equation is trivial. Ignore it.
- If all the coefficients of the unknowns are zero, and the right hand side is not, there is no way to satisfy it. So there is no solution to the given system of equations. Note so explicitly.
- In all other cases, solve the equation for the pivot variable (the variable with the pivot, i.e. with the first nonzero coefficient.) Take the other terms to the right hand side and substitute in anything you already learned in solving the previous equations below it.

4 LU Theorem

If you put minus the multipliers in a Gaussian elimination without partial pivoting in a lower triangular matrix L , with values 1 on the main diagonal, then

$$\boxed{LU = A} \tag{1}$$

That is the LU theorem.

After L and U have been found through Gaussian elimination, you can solve $A\vec{x} = \vec{b}$ for any given right hand side vector \vec{b} as follows: The system to solve is according to the theorem $LU\vec{x} = \vec{b}$. Temporarily call $U\vec{x} = \vec{y}$. Solve $L\vec{y} = \vec{b}$ using simple forward substitution. With \vec{y} now known, $U\vec{x} = \vec{y}$ can be solved using back substitution as before:

$$\boxed{L\vec{y} = \vec{b} \quad U\vec{x} = \vec{y}} \quad (2)$$

In case partial pivoting was needed, remember the row order interchanges you did and do the same interchanges with the coefficients of \vec{b} before solving as above. You also need to do the row exchanges on the forming matrix L while doing the GE. LU will then be the matrix A^{pp} , equal to matrix A except with its rows in the final order produced by all the row interchanges.

For large $n \times n$ matrices the number of computer “operations” (defined as 1 multiplication and 1 addition) to find L and U is approximately $\frac{1}{3}n^3$. After that, to find a \vec{x} given a \vec{b} takes only about n^2 operations.

Some warnings. Normally speaking:

1. Never ever use Cramer’s rule for anything but the tiniest of matrices. Multiplying out a determinant takes about $(n+1)!$ operations, which is gigantically larger than $\frac{1}{3}n^3$ for everything except the tiniest n . Not to mention the possible round-off error growth with so many operations. And the risk of overflow and underflow (i.e. numbers getting outside the range that they can be stored on a computer.)
2. If you find the inverse matrix A^{-1} , you can simply find any \vec{x} as $A^{-1}\vec{b}$. But that is normally a bad idea. One reason is that it takes n^3 operations to find A^{-1} , three times more than to find L and U . And to evaluate $A^{-1}\vec{b}$ still takes n^2 operations. Also the additional operations tend to increase round-off error, [1].
3. A *band matrix* is a matrix in which the nonzero elements are restricted to a relatively narrow band around the main diagonal. Never ever use an LU decomposition library routine designed for a full matrix to solve a system with a band matrix. The waste in storage and computer effort to store all these zeros, and compute with them, would be gigantic. Use an LU decomposition subroutine designed for a band matrix.
4. Never ever find the inverse of a band matrix if you can help it. The inverse will be a full matrix. However, L and U will still be band matrices. (Partial pivoting may increase U a bit to outside the band of A , by up to the width of L .)

5 Reduction to Row Canonical form

To reduce a matrix to “row canonical” form, (AKA “row reduced echelon” form, or “reduced row-echelon” form, or “Gauss-Jordan” form), first reduce it to echelon form using Gaussian elimination as described in section 1. Then, starting from the last equation with a nonzero coefficient of an unknown,

1. Use the equation to eliminate the first unknown with a nonzero coefficient in the equation, (i.e. the pivot unknown), from the previous equations (i.e. create zeros above the pivot). Do not worry about fractions anymore; they cannot normally be avoided.
2. Divide the equation by the pivot so that the pivot becomes 1.
3. Go to the previous equation, if any, and repeat the process.

Note: you can do both steps 1 and 2 above in each stage at the same time. You do not have to show each row nor the final scaling using a separate matrix. You must however explicitly show the multiplier(s) used.

Note: in this class you may *not* reduce a matrix to row canonical unless you are explicitly told so or the given class procedure requires it. Normally you must stop after you did the Gaussian elimination (producing normal echelon form with pivots that in general are not 1). That is because this is the standard and most efficient way to do it.

You may however proceed to row-canonical when finding the inverse of a matrix using the Gaussian Elimination method described later. You must proceed to row-canonical to find a simplified basis of the row space of a matrix, or of the column space, (starting from the transpose matrix), as described later.

6 Finding null spaces and solution spaces

To find the null space of a matrix, reduce it to echelon form as described earlier. To refresh your memory, the first nonzero elements in the rows of the echelon form are the pivots. Solve the homogeneous system by back substitution as also described earlier. To refresh your memory, you solve for the *pivot* variables. The variables *without* pivots cannot be solved for and become parameters with arbitrary values in the null space, multiplying “basis vectors”. The coefficients inside the basis vectors come from the solved variables or from writing trivialities.

For example, if your unknowns are $(x_1, x_2, x_3, x_4, x_5, x_6)$ and your echelon matrix is

$$\begin{pmatrix} 0 & 2 & 4 & 6 & 8 & 6 \\ 0 & 0 & 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 4 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

then the last equation is trivial and you get from the third

$$2x_5 + 4x_6 = 0 \implies x_5 = -2x_6$$

then from the second

$$x_4 = 0$$

then from the first

$$2x_2 + 4x_3 + 6(0) + 8(-2x_6) + 6x_6 = 0 \implies x_2 = -2x_3 + 5x_6$$

To get the null *space* (i.e. the full set of vectors $(x_1, x_2, x_3, x_4, x_5, x_6)$ that produce zero when premultiplied by the original A), the variables (x_1, x_3, x_6) *without pivots* go in the right hand side as arbitrary constants that can be anything:

$$\text{null space: } \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{pmatrix}_{\text{null space}} = x_1 \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + x_3 \begin{pmatrix} 0 \\ -2 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + x_6 \begin{pmatrix} 0 \\ 5 \\ 0 \\ 0 \\ -2 \\ 1 \end{pmatrix}$$

The coefficients for the pivot variables (x_2, x_4, x_5) in the vectors in the right hand side come from the solved equations, and those for (x_1, x_3, x_6) from trivialities.

To get a *basis* for the null space, you can use the constant vectors in the right hand side:

$$\text{a basis of the null space: } \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ -2 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \text{ and } \begin{pmatrix} 0 \\ 5 \\ 0 \\ 0 \\ -2 \\ 1 \end{pmatrix}$$

(By definition any vector in the null space is a linear combination of the above three vectors. And it is easy to see that the three are linearly independent.)

If the above basis would contain fractions, you should consider multiplying them by some nonzero constants to clean up. Note that that will of course affect the given expression for the null space.

To find a solution space is almost the same as finding the null space, except that you will use an augmented matrix to include the given nonzero right hand side. The right hand side will produce an additional vector in the solution space that is not multiplied by any unknown. Therefore the solution space is not a vector space (assuming that the given right hand side is not zero) and has no basis.

7 Finding inverses using GE

Note: if you are required to find an inverse using minors in this class, do not use *any* GE, even for attractive simplifications.

Note: If a matrix is orthonormal, its inverse *must* be found using a transpose in this class.

Finding the inverse of a matrix is usually a bad idea. But if you do need it, you may use the following trick:

1. Create an augmented matrix in which the n right hand side column vectors are the columns of the unit matrix.
2. Reduce the matrix to row canonical form using Gaussian elimination as described earlier. In other words

$$(A|I) \xrightarrow{\text{GE}} (A_{\text{RowCan}}|B_{\text{Red}})$$

3. For a nonsingular matrix A , A_{RowCan} will be a unit matrix and B_{Red} will be the desired A^{-1} . (For a singular matrix, one of the pivots will be found to be zero in the reduction to echelon form, and there is no inverse matrix.)

Note that what you are really doing in the above is solving the system of equations $AA^{-1} = I$ for the columns of A^{-1} . The only trick is that if you reduce the matrix all the way to a unit matrix, the solution A^{-1} becomes the same as the right hand side of the reduced system.

Also note that the most efficient way to find the inverse, in terms of operations, is still LU decomposition. Normally this would take $\frac{1}{3}n^3$ operations to find L and U and then n^3 operations to find the n columns of A^{-1} . But the right hand sides to the $U\vec{y} = \vec{b}$ systems are unit vectors, which are mostly zeros. If you use this to avoid computing coefficients in y that are automatically zero, you can reduce the operations by $\frac{1}{3}n^3$.

8 Finding inverses using minors

Note: if you are required to find an inverse using GE in this class, do not use *any* minors, even on 2×2 matrices.

Note: If a matrix is orthonormal, its inverse *must* be found using a transpose in this class.

In this class, if you need to find an inverse matrix using the method of minors,

- Write the transpose matrix.
- Form a matrix of the minors of each of the elements of the transpose matrix, with a checkerboard sign pattern. (Positive on the main diagonal.)

- Divide by the determinant.
- Evaluate.

9 Finding inverses using transposing

A matrix is orthonormal if its columns are mutually orthogonal and of length 1. For an orthonormal square matrix, its inverse *must* be found using a transpose in this class. This is mainly important, in this class, for the transformation matrices that diagonalize symmetric matrices.

10 Finding eigenvalues

In this class, to find the eigenvalues of an $n \times n$ matrix,

1. Form the matrix $A - \lambda I$. That means, add $-\lambda$ to each diagonal elements. (Don't forget zero diagonal elements.)
2. Find the determinant of that matrix using *the method of minors*. (Gaussian elimination is too messy here and should not be used.)
3. Set this determinant to zero. For an $n \times n$ matrix, the determinant can *always* be written in the form

$$|A| = \pm(\lambda - \lambda_1)(\lambda - \lambda_2) \dots (\lambda - \lambda_n)$$

where $\lambda_1, \lambda_2, \dots, \lambda_n$ are the *eigenvalues*. They are found as the roots (or zeros) of the determinant. There are always n eigenvalues. But these n eigenvalues do not necessarily correspond to n different numbers. For example, for some matrix λ_2 might be the same number as λ_4 . In that case, that number is called a “double eigenvalue”. If three of the eigenvalues are the same number, that number is called a “triple eigenvalue,” etcetera.

For an $n \times n$ matrix, the determinant is always a polynomial of degree n , call it $p_n(\lambda)$. Now finding the roots of quadratics, $n = 2$, is easy. But if the dimension of the matrix $n = 3$, you have to solve a cubic equation. For that the general solution is very and hard to apply, especially if you do not know complex variables. For $n = 4$, the general solution is even worse, and for $n = 5$ or more, there is no general expression for the roots at all. (It has in fact been proved that such an expression is impossible to find.) To deal with such problems, here are some tricks:

- Do not be too quick to multiply out. Maybe you can recognize a common factor $\lambda - \lambda_i$ in all terms before multiplying out. In that case, λ_i is one of your eigenvalues. To find the remaining eigenvalues, take the factor $\lambda - \lambda_i$ out of the entire expression and look at what is left.

- If you can guess a root λ_i (by trying, say, $0, \pm 1, \pm 2, \dots$, and seeing whether the determinant is zero for that value of λ), then you can write the original characteristic polynomial $p_n(\lambda)$ as $(\lambda - \lambda_i)p_{n-1}(\lambda)$ where $p_{n-1}(\lambda)$ is one degree less than $p_n(\lambda)$. So its roots are easier to find. You find $p_{n-1}(\lambda)$ by long division of $p_n(\lambda)$ by $(\lambda - \lambda_i)$.
- Sometimes a fourth order polynomial is really just a quadratic when written in terms of λ^2 , like $(\lambda^2)^2 - a(\lambda^2) - b$. In that case, first find the values of λ^2 , then from those the values of λ .

11 Eigenvectors of nonsymmetric matrices

To find the eigenvectors of a nonsymmetric matrix,

1. Find the eigenvalues of the matrix as described.
2. For each *distinct* eigenvalue λ_i , find the basis of the null space of $A - \lambda_i I$. This basis forms a complete set of eigenvectors \vec{e}_i for the eigenvalue. The method for finding nullspaces was discussed earlier.

There is always at least one eigenvector. However, for a multiple eigenvalue, the number of eigenvectors might be less than the multiplicity of the eigenvalue. If so, the matrix is *defective*. (This is not to be confused with singular. A matrix is singular if an eigenvalue is zero, making its determinant zero.) For example, a matrix is defective if a double eigenvalue has only one eigenvector. Or a triple eigenvalue has only one or two.

12 Eigenvectors of symmetric matrices

Real symmetric matrices (or more generally, complex Hermitian matrices) always have real eigenvalues, and they are never defective. Their eigenvectors can, and in this class must, be taken orthonormal. (Mutually orthogonal and of length 1.)

For real symmetric matrices, initially find the eigenvectors like for a nonsymmetric matrix. However, after that, you *must* make the eigenvectors orthonormal in this class.

- For eigenvectors belonging to single eigenvalues, orthogonality to all the other eigenvectors is automatic. Therefore it is enough to just divide the eigenvector by its length:

$$\vec{e}_i = \vec{e}_i^{\text{BAD}} / |\vec{e}_i^{\text{BAD}}| \quad (3)$$

Clean up, and that is it.

- However, eigenvectors belonging to a multiple eigenvalue are usually not mutually orthogonal if correctly found using the class procedure (as basis of the $A - \lambda_i I$ null space). You must *make* them orthogonal.

The most general method to do that is using the Gram-Schmidt procedure. This procedure works as follows. You start with the nonorthogonal eigenvectors corresponding to the multiple eigenvalue,

$$\vec{e}_1^{\text{BAD}}, \vec{e}_2^{\text{BAD}}, \vec{e}_3^{\text{BAD}}, \dots$$

obtained using the same procedure as for nonsymmetric matrices. For an eigenvalue of multiplicity m , there will be exactly m of these eigenvectors (since symmetric matrices can never be defective). I am assuming here that you have numbered these eigenvectors as 1, 2, \dots . If not, just replace 1, 2, \dots by i_1, i_2, \dots .

You create your first good eigenvector \vec{e}_1 as before, by simply dividing \vec{e}_1^{BAD} by its length:

$$\vec{e}_1 = \vec{e}_1^{\text{BAD}} / |\vec{e}_1^{\text{BAD}}|$$

Now you need to make the remaining vectors in the set of m orthogonal to \vec{e}_1 . You do that by removing from each vector its vector component in the direction of \vec{e}_1 , as follows:

$$\vec{e}_2^{\text{BAD},2} = \vec{e}_2^{\text{BAD}} - (\vec{e}_2^{\text{BAD}} \cdot \vec{e}_1)\vec{e}_1 \quad \vec{e}_3^{\text{BAD},2} = \vec{e}_3^{\text{BAD}} - (\vec{e}_3^{\text{BAD}} \cdot \vec{e}_1)\vec{e}_1 \quad \dots$$

Vectors $\vec{e}_2^{\text{BAD},2}, \vec{e}_3^{\text{BAD},2}, \dots$ are now orthogonal to \vec{e}_1 .

Now you create your second good eigenvector \vec{e}_2 by simply dividing $\vec{e}_2^{\text{BAD},2}$ by its length

$$\vec{e}_2 = \vec{e}_2^{\text{BAD},2} / |\vec{e}_2^{\text{BAD},2}|$$

If there are still vectors left, you need to make each orthogonal to \vec{e}_2 . In the same way as before

$$\vec{e}_3^{\text{BAD},3} = \vec{e}_3^{\text{BAD},2} - (\vec{e}_3^{\text{BAD},2} \cdot \vec{e}_2)\vec{e}_2 \quad \dots$$

Now you create your third good eigenvector \vec{e}_3 by simply dividing $\vec{e}_3^{\text{BAD},3}$ by its length

$$\vec{e}_3 = \vec{e}_3^{\text{BAD},3} / |\vec{e}_3^{\text{BAD},3}|$$

If there are still vectors $\vec{e}_4^{\text{BAD},3}, \dots$ left, you now have to make them orthogonal to \vec{e}_3 . Then you can find \vec{e}_4 by dividing $\vec{e}_4^{\text{BAD},4}$ by its length.

Continue like this for multiplicities greater than 4. The general formula is

$$\boxed{\vec{e}_j^{\text{BAD,new}} = \vec{e}_j^{\text{BAD,old}} - (\vec{e}_j^{\text{BAD,old}} \cdot \vec{e}_i)\vec{e}_i \quad \text{for } i = j+1, j+2, \dots, m} \quad (4)$$

where e_i is the last good eigenvector obtained so far and $j > i$.

13 Change of basis (coordinate system)

Normally, you use “Cartesian coordinates” when dealing with vectors. That is based on “Cartesian basis vectors” $\hat{i}, \hat{j}, \hat{k}, \dots$. For example, in two dimensions,

$$\hat{i} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \hat{j} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

These Cartesian basis vectors are shown in red in figure 1(a). Using these two basis vectors, you can write *any* vector \vec{v} in the form

$$\vec{v} = v_1 \hat{i} + v_2 \hat{j}$$

This is illustrated graphically in figure 1(b). Note how, starting from the origin, if you first move over $v_1 \hat{i}$ and then over $v_2 \hat{j}$ you reach the end point of vector \vec{v} . Now v_1 and v_2 are *ordinary numbers* that are called the “components”, or “coefficients,” or “coordinates” of vector \vec{v} .

Therefore, in this way, you never have to deal with more vectors than \hat{i} and \hat{j} . All the rest is just ordinary numbers.

But sometimes it is convenient to use a different basis than the obvious \hat{i}, \hat{j} one. For example, you might know that in dealing with plane stresses, it is often convenient to rotate the coordinate system to the “principal axes.” In principal axes, there are no shear stresses, just normal stresses. But if you rotate the coordinate system, \hat{i} and \hat{j} become different vectors, call them \hat{i}' and \hat{j}' . The point however is that in using these *new* basis vectors \hat{i}' and \hat{j}' , your physical problem has simplified.

As we will see later, to simplify a problem, the desired new vectors are not always orthonormal (orthogonal and of length 1) like \hat{i}' and \hat{j}' in the example above. In general, the new basis vectors, we will call them \vec{p}_1 and \vec{p}_2 , can be anything, as long as they are linearly independent. As long as that is true, you can still write any arbitrary vector \vec{v} as

$$\vec{v} = v'_1 \vec{p}_1 + v'_2 \vec{p}_2$$

where v'_1 and v'_2 are still ordinary numbers. They are called the coordinates of \vec{v} in the new basis \vec{p}_1, \vec{p}_2 . It is illustrated graphically in figure 1(c).

However, in general the coordinates v'_1 and v'_2 in the new basis are not the same as the coordinates v_1 and v_2 in the old basis \hat{i}, \hat{j} . So, if you want to use the new basis to your advantage, you will normally have to know how to compute v'_1 and v'_2 if you know v_1 and v_2 and/or vice-versa. That is the problem that this section will address.

First, to find the old coordinates v_1 and v_2 given the new ones v'_1 and v'_2 is easy. Just write the above equation as a row-column multiplication:

$$\vec{v} = \begin{pmatrix} \vec{p}_1 & \vec{p}_2 \end{pmatrix} \begin{pmatrix} v'_1 \\ v'_2 \end{pmatrix}$$

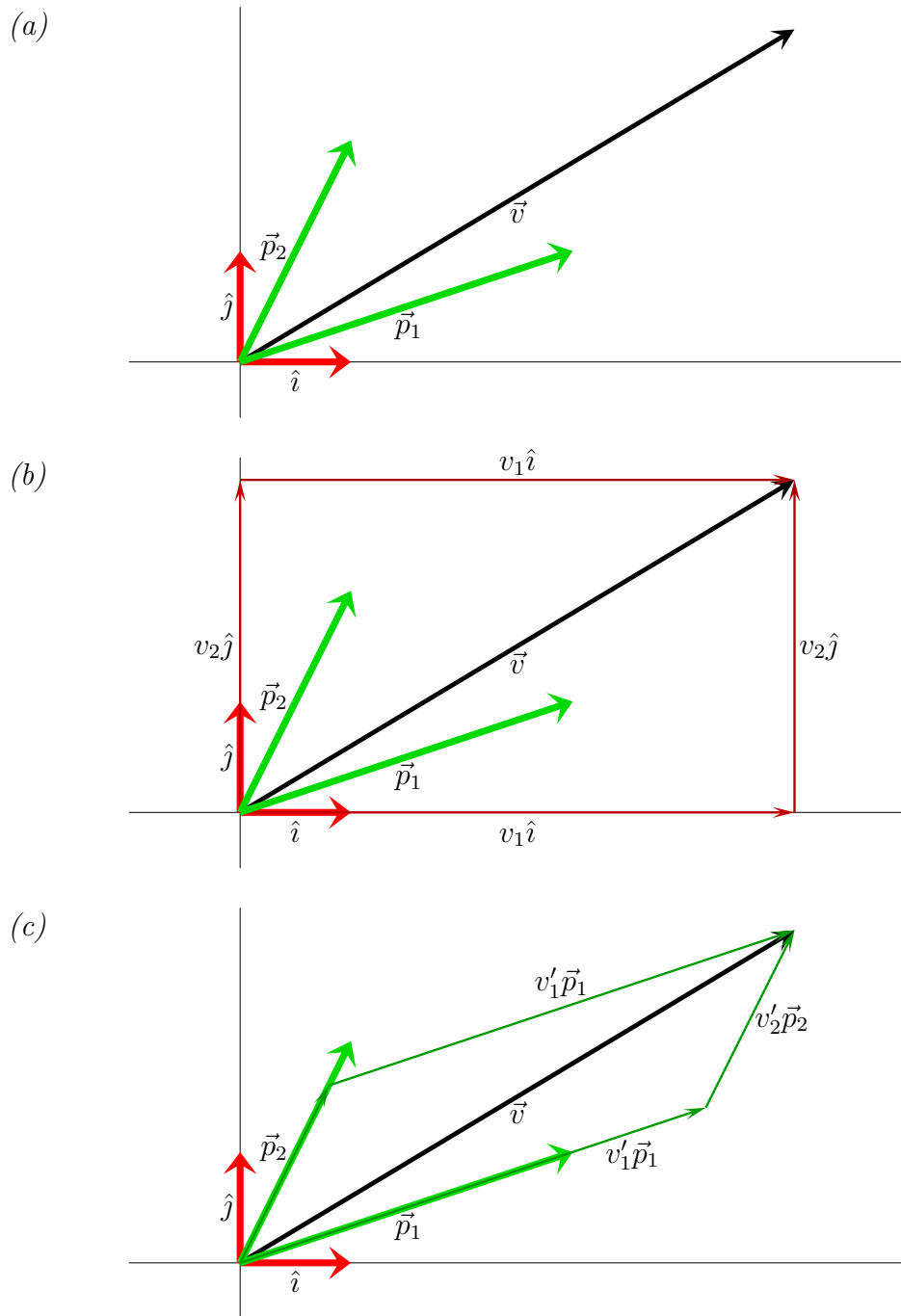


Figure 1: Illustration of a change of basis in two dimensions. (a) The standard basis \hat{i}, \hat{j} , a new basis \vec{p}_1, \vec{p}_2 , and any arbitrary vector \vec{v} . (b) Any vector \vec{v} can be written as $\vec{v} = v_1\hat{i} + v_2\hat{j}$. (c) However, the same vector can also be written as $\vec{v} = v'_1\vec{p}_1 + v'_2\vec{p}_2$. That uses the new basis.

Then write that out in terms of the old Cartesian coordinates:

$$\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} \vec{p}_1 & \vec{p}_2 \end{pmatrix} \begin{pmatrix} v'_1 \\ v'_2 \end{pmatrix}$$

or more concisely,

$$\vec{v} = \begin{pmatrix} \vec{p}_1 & \vec{p}_2 \end{pmatrix} \vec{v}'$$

where a prime on a vector means that its coordinates are written out in terms of the new basis, and no prime means they are written out in terms of the old basis. According to the above, to get the old coordinates from the new ones, just put \vec{p}_1 and \vec{p}_2 in a matrix, call it P ,

$$\boxed{P = \begin{pmatrix} \vec{p}_1 & \vec{p}_2 \end{pmatrix}} \quad (5)$$

and multiply by that matrix. Do note that the \vec{p}_1 and \vec{p}_2 that you put in P must be written out in terms of the old Cartesian coordinates. But what else could you do?

So you get the following transformation formulae between coordinates

$$\boxed{\vec{v} = P\vec{v}' \quad \vec{v}' = P^{-1}\vec{v}} \quad (6)$$

with P as above. Note that while P computes the *old* coordinates *from* the *new* ones, it is called “the transformation matrix from *old to new*”. It does not make any sense, but that is what mathematicians call it. Just remember, “*old to new*” really means *old from new*.

The final thing you need to know is what happens to matrices. If a matrix A converts a vector \vec{v} to a vector \vec{w} in terms of Cartesian coordinates, then A' should convert \vec{v}' to \vec{w}' in terms of the new coordinates. Since

$$A\vec{v} = \vec{w} \implies AP\vec{v}' = P\vec{w}' \implies P^{-1}AP\vec{v}' = \vec{w}'$$

the desired matrix A' is seen to equal $P^{-1}AP$.

So the transformation rules for matrices are

$$\boxed{A = PA'P^{-1} \quad A' = P^{-1}AP} \quad (7)$$

That is much like the ones for vectors, except there is an additional trailing P^{-1} , respectively P .

While we used a two dimensional example, you can do all of the above in any number of dimensions. You just add more basis vectors to transformation matrix P .

14 Diagonalization of nonsymmetric matrices

If a matrix A is not defective, you can use its eigenvectors as new basis. It turns out that in that basis the matrix simplifies to a *diagonal* matrix

$$A' = \begin{pmatrix} \lambda_1 & 0 & 0 & \cdots \\ 0 & \lambda_2 & 0 & \cdots \\ 0 & 0 & \lambda_3 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \quad (8)$$

Since this diagonal matrix has the eigenvalues on the main diagonal, (in the order that you arranged the corresponding eigenvectors), it is often written as Λ instead of A' .

Needless to say, this simplification is a tremendous help if you are doing analytical or numerical work involving the matrix.

The quickest way to see why A' is diagonal like above is to note that in terms of the new basis, $A\vec{v}$ produces a new vector \vec{w}' according to

$$\vec{w}' = A'\vec{v}' = A'(v'_1\vec{e}_1 + v'_2\vec{e}_2 + \dots + v'_n\vec{e}_n) = v'_1\lambda_1\vec{e}_1 + v'_2\lambda_2\vec{e}_2 + \dots + v'_n\lambda_n\vec{e}_n$$

since $\vec{e}_1, \vec{e}_2, \dots$ are eigenvectors of A . So the coefficients of \vec{w}' are related to those of \vec{v}' as $v'_1\lambda_1, v'_2\lambda_2, \dots, v'_n\lambda_n$. And that is just what multiplying by the diagonal matrix A' above accomplishes.

Recall from section 13 that the transformation matrix P for change of basis to the eigenvectors must equal the matrix E of eigenvectors. You therefore have for any vector \vec{v} and matrix M that you want to transform from new coordinates to old or vice-versa:

$$\vec{v} = E\vec{v}' \quad \vec{v}' = E^{-1}\vec{v} \quad M = EM'E^{-1} \quad M' = E^{-1}ME \quad (9)$$

Here the primes mean the vector or matrix as it appears in the new basis of eigenvectors.

In summary, a nondefective matrix becomes diagonal when its eigenvectors are used as basis. The main diagonal contains the eigenvalues, ordered like the corresponding eigenvectors in E .

15 Diagonalization of symmetric matrices

For symmetric matrices the same observations apply as for nonsymmetric matrices in the previous section. But there are some further considerations.

Most importantly, the eigenvectors, if found using the class procedure, are orthonormal. So you can consider the eigenvectors $\vec{e}_1, \vec{e}_2, \dots$ to be a rotated *Cartesian* basis. To make this clearer to other people, you should rename $\vec{e}_1, \vec{e}_2, \dots$ to $\hat{i}', \hat{j}', \dots$

The most important other thing to remember is that the transformation matrix

$$E = (\hat{i}' \hat{j}' \dots)$$

is now orthonormal. So be sure to use section 9 to find its inverse.

Note further that if the determinant of the transformation matrix E is negative, the rotated coordinate system is also left-handed instead of right-handed. It corresponds to a mirroring of a coordinate besides the rotation. If this bothers you, multiply one of the eigenvectors by minus one.

References

- [1] G. Dahlquist and Å. Björck. *Numerical Methods*. Prentice-Hall, 1974. Translated by Ned Anderson.
- [2] Dennis G. Zill and Warren S. Wright. *Advanced Engineering Mathematics*. Jones and Bartlett Learning, 5th edition, 2014.