

10.8: H- Tutorial on Matrix Diagonalization

You know from as far back as your introductory mechanics course that some problems are difficult given one choice of coordinate axes and easy or even trivial given another. (For example, the famous “monkey and hunter” problem is difficult using a horizontal axis, but easy using an axis stretching from the hunter to the monkey.) The mathematical field of linear algebra is devoted, in large part, to systematic techniques for finding coordinate systems that make problems easy. This tutorial introduces the most valuable of these techniques. It assumes that you are familiar with matrix multiplication and with the ideas of the inverse, the transpose, and the determinant of a square matrix. It is also useful to have a nodding acquaintance with the inertia tensor.

This presentation is intentionally non-rigorous. A rigorous, formal treatment of matrix diagonalization can be found in any linear algebra textbook,¹ and there is no need to duplicate that function here. What is provided here instead is a heuristic picture of what’s going on in matrix diagonalization, how it works, and why anyone would want to do such a thing anyway. Thus this presentation complements, rather than replaces, the logically impeccable (“bulletproof”) arguments of the mathematics texts.

Essential problems in this tutorial are marked by asterisks (*).

Warning: This tutorial is still in **draft** form.

H.1 What’s in a name?

There is a difference between an entity and its name. For example, a tree is made of wood, whereas its name “tree” made of ink. One way to see this is to note that in German, the name for a tree is “Baum”, so the name changes upon translation, but the tree itself does not change. (Throughout this tutorial, the term “translate” is used as in “translate from one language to another” rather than as in “translate by moving in a straight line”.)

The same holds for mathematical entities. Suppose a length is represented by the number “2” because it is two feet long. Then the same length is represented by the number “24” because it is twenty-four inches long. The same length is represented by two different numbers, just as the same tree has two different names. The representation of a length as a number depends not only upon the length, but also upon the coordinate system used to measure the length.

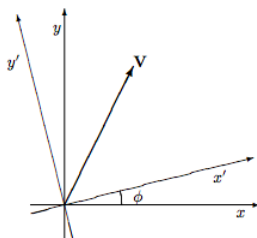
H.2 Vectors in two dimensions

One way of describing a two-dimensional vector \mathbf{V} is by giving its x and y components in the form of a 2×1 column matrix

$$\begin{pmatrix} V_x \\ V_y \end{pmatrix}. \quad (10.8.1)$$

Indeed it is sometimes said that the vector \mathbf{V} is *equal* to the column matrix (H.1). This is not precisely correct—it is better to say that the vector is *described by* the column matrix or *represented by* the column matrix or that its *name* is the column matrix. This is because if you describe the vector using a different set of coordinate axes you will come up with a different column matrix to describe the same vector. For example, in the situation shown below the descriptions in terms of the two different coordinate systems are related through the matrix equation

$$\begin{pmatrix} V_{x'} \\ V_{y'} \end{pmatrix} = \begin{pmatrix} \cos \phi & \sin \phi \\ -\sin \phi & \cos \phi \end{pmatrix} \begin{pmatrix} V_x \\ V_y \end{pmatrix}. \quad (10.8.2)$$



The 2×2 matrix above is called the “rotation matrix” and is usually denoted by $R(\phi)$:

$$R(\phi) \equiv \begin{pmatrix} \cos \phi & \sin \phi \\ -\sin \phi & \cos \phi \end{pmatrix} \quad (10.8.3)$$

One interesting property of the rotation matrix is that it is always invertible, and that its inverse is equal to its transpose. Such matrices are called *orthogonal*.¹ You could prove this by working a matrix multiplication, but it is easier to simply realize that the inverse of a rotation by ϕ is simply a rotation by $-\phi$, and noting that

$$\mathbf{R}^{-1}(\phi) = \mathbf{R}(-\phi) = \mathbf{R}^{\dagger}(\phi). \quad (10.8.4)$$

(The dagger represents matrix transposition.)

There are, of course, an infinite number of column matrix representations for any vector, corresponding to the infinite number of coordinate axis rotations with ϕ from 0 to 2π . But one of these representations is special: It is the one in which the x' -axis lines up with the vector, so the column matrix representation is just

$$\begin{pmatrix} V \\ 0 \end{pmatrix}, \quad (10.8.5)$$

where $V = |\mathbf{V}| = \sqrt{V_x^2 + V_y^2}$ is the magnitude of the vector. This set of coordinates is the preferred (or “canonical”) set for dealing with this vector: one of the two components is zero, the easiest number to deal with, and the other component is a physically important number. You might wonder how I can claim that this representation has full information about the vector: The initial representation (H.1) contains two independent numbers, whereas the preferred representation (H.5) contains only one. The answer is that the preferred representation contains one number (the magnitude of the vector) explicitly while another number (the polar angle of the vector relative to the initial x -axis) is contained implicitly in the rotation needed to produce the preferred coordinate system.

H.1 Problem: Right angle rotations

Verify equation (H.2) in the special cases $\phi = 90^\circ$, $\phi = 180^\circ$, $\phi = 270^\circ$, and $\phi = 360^\circ$.

H.2 Problem: The rotation matrix

a. Derive equation (H.2) through purely geometrical arguments.

b. Express $\hat{\mathbf{i}}'$ and $\hat{\mathbf{j}}'$, the unit vectors of the (x', y') coordinate system, as linear combinations of $\hat{\mathbf{i}}$ and $\hat{\mathbf{j}}$. Then use

$$V_{x'} = \mathbf{V} \cdot \hat{\mathbf{i}}' \quad \text{and} \quad V_{y'} = \mathbf{V} \cdot \hat{\mathbf{j}}' \quad (10.8.6)$$

to derive equation (H.2).

c. Which derivation do you find easier?

H.3 Problem: Rotation to the preferred coordinate system*

In the preferred coordinate system, $V_{y'} = 0$. Use this requirement to show that the preferred system is rotated from the initial system by an angle ϕ with

$$\tan \phi = \frac{V_y}{V_x}. \quad (10.8.7)$$

For any value of V_y/V_x , there are two angles that satisfy this equation. What is the representation of \mathbf{V} in each of these two coordinate systems?

H.4 Problem: A non-rotation orthogonal transformation

In one coordinate system the y -axis is vertical and the x -axis points to the right. In another the y' -axis is vertical and the x' -axis points to the left. Find the matrix that translates vector coordinates from one system to the other. Show that this matrix is orthogonal but not a rotation matrix.

H.5 Problem: Other changes of coordinate*

Suppose vertical distances (distances in the y direction) are measured in feet while horizontal distances (distances in the x direction) are measured in miles. (This system is not perverse. It is used in nearly all American road maps.) Find the matrix that changes the representation of a vector in this coordinate system to the representation of a vector in a system where all distances are measured in feet. Find the matrix that translates back. Are these matrices orthogonal?

H.6 Problem: Other special representations

At equation (H.5) we mentioned one “special” (or “canonical”) representation of a vector. There are three others, namely

$$\begin{pmatrix} 0 \\ -V \end{pmatrix}, \quad \begin{pmatrix} -V \\ 0 \end{pmatrix}, \quad \begin{pmatrix} 0 \\ V \end{pmatrix}. \quad (10.8.8)$$

If coordinate-system rotation angle ϕ brings the vector representation into the form (H.5), then what rotation angle will result in these three representations?

H.3 Tensors in two dimensions

A tensor, like a vector, is a geometrical entity that may be described (“named”) through components, but a d -dimensional tensor requires d^2 rather than d components. Tensors are less familiar and more difficult to visualize than vectors, but they are neither less important nor “less physical”. We will introduce tensors through the concrete example of the inertia tensor of classical mechanics (see, for example, reference [2]), but the results we present will be perfectly general.

Just as the two components of a two-dimensional vector are most easily kept track of through a 2×1 matrix, so the four components of two-dimensional tensor are most conveniently written in the form of a 2×2 matrix. For example, the inertia tensor \mathbf{T} of a point particle with mass m located² at (x, y) has components

(Note the distinction between the tensor \mathbf{T} and its matrix of components, its “name”, T .) As with vector components, the tensor components are different in different coordinate systems, although the tensor itself does not change. For example, in the primed coordinate system of the figure on page 231, the tensor components are of course

$$T' = \begin{pmatrix} my'^2 & -mx'y' \\ -mx'y' & mx'^2 \end{pmatrix}. \quad (10.8.9)$$

A little calculation shows that the components of the inertia tensor in two different coordinate systems are related through

$$T' = R(\phi)TR^{-1}(\phi). \quad (10.8.10)$$

This relation holds for any tensor, not just the inertia tensor. (In fact, one way to define “tensor” is as an entity with four components that satisfy the above relation under rotation.) If the matrix representing a tensor is symmetric (i.e. the matrix is equal to its transpose) in one coordinate system, then it is symmetric in all coordinate systems (see problem H.7). Therefore the symmetry is a property of the tensor, not of its matrix representation, and we may speak of “a symmetric tensor” rather than just “a tensor represented by a symmetric matrix”.

As with vectors, one of the many matrix representations of a given tensor is considered special (or “canonical”): It is the one in which the lower left component is zero. Furthermore if the tensor is symmetric (as the inertia tensor is) then in this preferred coordinate system the upper right component will be zero also, so the matrix will be all zeros except for the diagonal elements. Such a matrix is called a “diagonal matrix” and the process of finding the rotation that renders the matrix representation of a symmetric tensor diagonal is called “diagonalization”.³ We may do an “accounting of information” for this preferred coordinate system just as we did with vectors. In the initial coordinate system, the symmetric tensor had three independent components. In the preferred system, it has two independent components manifestly visible in the diagonal matrix representation, and one number hidden through the specification of the rotation.

H.7 Problem: Representations of symmetric tensors*

Show that if the matrix S representing a tensor is symmetric, and if B is any orthogonal matrix, then all of the representations

$$BSB^\dagger \quad (10.8.11)$$

are symmetric. (Clue: If you try to solve this problem for rotations in two dimensions using the explicit rotation matrix (H.3), you will find it solvable but messy. The clue is that this problem asks you do prove the result in any number of dimensions, and for any orthogonal matrix B , not just rotation matrices. This more general problem is considerably easier to solve.)

H.8 Problem: Diagonal inertia tensor

The matrix (H.9) represents the inertia tensor of a point particle with mass m located a distance r from the origin. Show that the matrix is diagonal in four different coordinate systems: one in which the x' -axis points directly toward the particle, one in which the y' -axis points directly away from the particle, one in which the x' -axis points directly away from the particle, and

one in which the y' -axis points directly toward the particle. Find the matrix representation in each of these four coordinate systems.

H.9 Problem: Representations of a certain tensor

Show that a tensor represented in one coordinate system by a diagonal matrix with equal elements, namely

$$\begin{pmatrix} d_0 & 0 \\ 0 & d_0 \end{pmatrix}, \quad (10.8.12)$$

has the same representation in all orthogonal coordinate systems.

H.10 Problem: Rotation to the preferred coordinate system*

A tensor is represented in the initial coordinate system by

$$\begin{pmatrix} a & b \\ b & c \end{pmatrix}. \quad (10.8.13)$$

Show that the tensor is diagonal in a preferred coordinate system which is rotated from the initial system by an angle ϕ with

$$\tan(2\phi) = \frac{2b}{a-c}. \quad (10.8.14)$$

This equation has four solutions. Find the rotation matrix for $\phi = 90^\circ$, then show how the four different diagonal representations are related. You do not need to find any of the diagonal representations in terms of a , b and c . . . just show what the other three are given that one of them is

$$\begin{pmatrix} d_1 & 0 \\ 0 & d_2 \end{pmatrix}. \quad (10.8.15)$$

H.11 Problem: Inertia tensor in outer product notation

The discussion in this section has emphasized the tensor's matrix representation ("name") \mathbf{T} rather than the tensor \mathbf{T} itself.

a. Define the "identity tensor" $\mathbf{1}$ as the tensor represented in some coordinate system by

$$\mathbf{1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \quad (10.8.16)$$

Show that this tensor has the same representation in any coordinate system.

b. Show that the inner product between two vectors results in a scalar: Namely

if vector \mathbf{b} is represented by $\begin{pmatrix} a_x \\ a_y \end{pmatrix}$ and vector \mathbf{a} is represented by $\begin{pmatrix} b_x \\ b_y \end{pmatrix}$

then the inner product $\mathbf{a} \cdot \mathbf{b}$ is given through

$$\begin{pmatrix} a_x & a_y \end{pmatrix} \begin{pmatrix} b_x \\ b_y \end{pmatrix} = a_x b_x + a_y b_y$$

and this inner product is a scalar. (A 1×2 matrix times a 2×1 matrix is a 1×1 matrix.) That is, the vector \mathbf{a} is represented by different coordinates in different coordinate systems, and the vector \mathbf{b} is represented by different coordinates in different coordinate systems, but the inner product $\mathbf{a} \cdot \mathbf{b}$ is the same in all coordinate systems.

c. In contrast, show that the outer product of two vectors is a tensor: Namely

$$\mathbf{a} \mathbf{b} \doteq \begin{pmatrix} a_x \\ a_y \end{pmatrix} \begin{pmatrix} b_x & b_y \end{pmatrix} = \begin{pmatrix} a_x b_x & a_x b_y \\ a_y b_x & a_y b_y \end{pmatrix}$$

(A 2×1 matrix times a 1×2 matrix is a 2×2 matrix.) That is, show that the representation of $\mathbf{a} \mathbf{b}$ transforms from one coordinate system to another as specified through (H.11).

d. Show that the inertia tensor for a single particle of mass m located at position \mathbf{r} can be written in coordinate-independent fashion as

$$\mathbf{T} = m\mathbf{1}r^2 - m\mathbf{r}\mathbf{r}. \quad (10.8.17)$$

H.4 Tensors in three dimensions

A three-dimensional tensor is represented in component form by a 3×3 matrix with nine entries. If the tensor is symmetric, there are six independent elements. . . three on the diagonal and three off-diagonal. The components of a tensor in three dimensions change with coordinate system according to

$$\mathbf{T}' = \mathbf{R}\mathbf{T}\mathbf{R}^\dagger, \quad (10.8.18)$$

where \mathbf{R} is the 3×3 rotation matrix.

A rotation in two dimension is described completely by giving a single angle. In three dimensions more information is required. Specifically, we need not only the amount of the rotation, but we must also know the plane in which the rotation takes place. We can specify the plane by giving the unit vector perpendicular to that plane. Specifying an arbitrary vector in three dimensions requires three numbers, but specifying a unit vector in three dimensions requires only two numbers because the magnitude is already fixed at unity. Thus three numbers are required to specify a rotation in three dimensions: two to specify the rotation's plane, one to specify the rotation's size. (One particularly convenient way to specify a three-dimensional rotation is through the three *Euler angles*. Reference [3] defines these angles and shows how to write the 3×3 rotation matrix in terms of these variables. For the purposes of this tutorial, however, we will not need an explicit rotation matrix. . . all we need is to know is the number of angles required to specify a rotation.)

In two dimensions, any symmetric tensor (which has three independent elements), could be represented by a diagonal tensor (with two independent elements) plus a rotation (one angle). We were able to back up this claim with an explicit expression for the angle.

In three dimensions it seems reasonable that any symmetric tensor (six independent elements) can be represented by a diagonal tensor (three independent elements) plus a rotation (three angles). The three angles just have to be selected carefully enough to make sure that they cause the off-diagonal elements to vanish. This supposition is indeed correct, although we will not pause for long enough to prove it by producing explicit formulas for the three angles.

H.5 Tensors in d dimensions

A d -dimensional tensor is represented by a $d \times d$ matrix with d^2 entries. If the tensor is symmetric, there are d independent on-diagonal elements and $d(d-1)/2$ independent off-diagonal elements. The tensor components will change with coordinate system in the now-familiar form

$$\mathbf{T}' = \mathbf{R}\mathbf{T}\mathbf{R}^\dagger, \quad (10.8.19)$$

where \mathbf{R} is the $d \times d$ rotation matrix.

How many angles does it take to specify a rotation in d dimensions? Remember how we went from two dimensions to three: The three dimensional rotation took place “in a plane”, i.e. in a two-dimensional subspace. It required two (i.e. $d - 1$) angles to specify the orientation of the plane plus one to specify the rotation within the plane. . . a total of three angles.

A rotation in four dimensions takes place within a three-dimensional subspace. It requires $3 = d - 1$ angles to specify the orientation of the three-dimensional subspace, plus, as we found above, three angles to specify the rotation within the three-dimensional subspace. . . a total of six angles.

A rotation in five dimensions requires $4 = d - 1$ angles to specify the four-dimensional subspace in which the rotation occurs, plus the six angles that we have just found specify a rotation within that subspace. . . a total of ten angles.

In general, the number of angles needed to specify a rotation in d dimensions is

$$A_d = d - 1 + A_{d-1} = d(d-1)/2. \quad (10.8.20)$$

This is exactly the number of independent off-diagonal elements in a symmetric tensor. It seems reasonable that we can choose the angles to ensure that, in the resulting coordinate system, all the off-diagonal elements vanish. The proof of this result is difficult and proceeds in a very different manner from the plausibility argument sketched here. (The proof involves concepts like eigenvectors and eigenvalues, and it gives an explicit recipe for constructing the rotation matrix. It has the advantage of rigor and the disadvantage of being so technical that it's easy to lose track of the fact that that all you're doing is choosing a coordinate system.)

H.12 Problem: Non-symmetric tensors*

Argue that a non-symmetric tensor can be brought into a “triangular” representation in which all the elements below the diagonal are equal to zero and all the elements on and above the diagonal are independent. (This is indeed the case, although in general some of the non-zero elements remaining will be complex-valued, and some of the angles will involve rotations into complex-valued vectors.)

H.6 Linear transformations in two dimensions

Section H.3 considered 2×2 matrices as representations of tensors. This section gains additional insight by considering 2×2 matrices as representations of linear transformations. It demonstrates how diagonalization can be useful and gives a clue to an efficient algorithm for diagonalization.

A linear transformation is a function from vectors to vectors that can be represented in any given coordinate system as

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}. \quad (10.8.21)$$

If the equation above represents (“names”) the transformation in one coordinate system, what is its representation in some other coordinate system? We assume that the two coordinate systems are related through an orthogonal matrix B such that

$$\begin{pmatrix} u' \\ v' \end{pmatrix} = B \begin{pmatrix} u \\ v \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} x' \\ y' \end{pmatrix} = B \begin{pmatrix} x \\ y \end{pmatrix}. \quad (10.8.22)$$

(For example, if the new coordinate system is the primed coordinate system of the figure on page 231, then the matrix B that translates from the original to the new coordinates is the rotation matrix $R(\phi)$.) Given this “translation dictionary”, we have

$$\begin{pmatrix} u' \\ v' \end{pmatrix} = B \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}. \quad (10.8.23)$$

But B is invertible, so

$$\begin{pmatrix} x \\ y \end{pmatrix} = B^{-1} \begin{pmatrix} x' \\ y' \end{pmatrix} \quad (10.8.24)$$

whence

$$\begin{pmatrix} u' \\ v' \end{pmatrix} = B \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} B^{-1} \begin{pmatrix} x' \\ y' \end{pmatrix}. \quad (10.8.25)$$

Thus the representation of the transformation in the primed coordinate system is

$$B \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} B^{-1} \quad (10.8.26)$$

(compare equation H.11). This equation has a very direct physical meaning. Remember that the matrix B translates from the old (x, y) coordinates to the new (x', y') coordinates, while the matrix B^{-1} translates in the opposite direction. Thus the equation above says that the representation of a transformation in the new coordinates is given by translating from new to old coordinates (through the matrix B^{-1}), then applying the old representation (the “a matrix”) to those old coordinates, and finally translating back from old to new coordinates (through the matrix B).

The rest of this section considers only transformations represented by symmetric matrices, which we will denote by

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} a & b \\ b & c \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}. \quad (10.8.27)$$

Let’s try to understand this transformation as something more than a jumble of symbols awaiting a plunge into the calculator. First of all, suppose the vector \mathbf{V} maps to the vector \mathbf{W} . Then the vector $5\mathbf{V}$ will be mapped to vector $5\mathbf{W}$. In short, if we know how the transformation acts on vectors with magnitude unity, we will be able to see immediately how it acts on vectors with other magnitudes. Thus we focus our attention on vectors on the unit circle:

$$x^2 + y^2 = 1. \quad (10.8.28)$$

A brief calculation shows that the length of the output vector is then

$$\sqrt{u^2 + v^2} = \sqrt{a^2 x^2 + b^2 + c^2 y^2 + 2b(a+c)xy}, \quad (10.8.29)$$

which isn't very helpful. Another brief calculation shows that if the input vector has polar angle θ , then the output vector has polar angle ϕ with

$$\tan \phi = \frac{b + c \tan \theta}{a + b \tan \theta}, \quad (10.8.30)$$

which is similarly opaque and messy.

Instead of trying to understand the transformation in its initial coordinate system, let's instead convert (rotate) to the special coordinate system in which the transformation is represented by a diagonal matrix. In this system,

$$\begin{pmatrix} u' \\ v' \end{pmatrix} = \begin{pmatrix} d_1 & 0 \\ 0 & d_2 \end{pmatrix} \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} d_1 x' \\ d_2 y' \end{pmatrix}. \quad (10.8.31)$$

The unit circle is still

$$x'^2 + y'^2 = 1, \quad (10.8.32)$$

so the image of the unit circle is

$$\left(\frac{u'}{d_1}\right)^2 + \left(\frac{v'}{d_2}\right)^2 = 1, \quad (10.8.33)$$

namely an ellipse! This result is transparent in the special coordinate system, but almost impossible to see in the original one.

Note particularly what happens to a vector pointing along the x' coordinate axis. For example, the unit vector in this direction transforms to

$$\begin{pmatrix} d_1 \\ 0 \end{pmatrix} = \begin{pmatrix} d_1 & 0 \\ 0 & d_2 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix}. \quad (10.8.34)$$

In other words, when the vector is transformed it changes in magnitude, but not in direction. Vectors with this property are called eigenvectors. It is easy to see that any vector on either the x' or y' coordinate axes are eigenvectors.

H.7 What does “eigen” mean?

If a vector \mathbf{x} is acted upon by a linear transformation \mathbf{B} , then the output vector

$$\mathbf{x}' = \mathbf{B}\mathbf{x} \quad (10.8.35)$$

will usually be skew to the original vector \mathbf{x} . However, for some very special vectors it might just happen that \mathbf{x}' is parallel to \mathbf{x} . Such vectors are called “eigenvectors”. (This is a terrible name because (1) it gives no idea of what eigenvectors are or why they're so important and (2) it sounds gross. However, that's what they're called.) We have already seen, in the previous section, that eigenvectors are related to coordinate systems in which the transformation is particularly easy to understand. If \mathbf{x} is an eigenvector, then

$$\mathbf{B}\mathbf{x} = \lambda\mathbf{x}, \quad (10.8.36)$$

where λ is a scalar called “the eigenvalue associated with eigenvector \mathbf{x} ”. If \mathbf{x} is an eigenvector, then any vector parallel to \mathbf{x} is also an eigenvector with the same eigenvalue. (That is, any vector of the form $c\mathbf{x}$, where c is any scalar, is also an eigenvector with the same eigenvalue.) Sometimes we speak of a “line of eigenvectors”.

The vector $\mathbf{x} = \mathbf{0}$ is never considered an eigenvector, because

$$\mathbf{B}\mathbf{0} = \lambda\mathbf{0}, \quad (10.8.37)$$

for any value of λ for any linear transformation. On the other hand, if

$$\mathbf{B}\mathbf{x} = \mathbf{0}\mathbf{x} = \mathbf{0} \quad (10.8.38)$$

for some non-zero vector \mathbf{x} , then \mathbf{x} is an eigenvector with eigenvalue $\lambda = 0$.

H.13 Problem: Plane of eigenvectors

Suppose \mathbf{x} and \mathbf{y} are two non-parallel vectors with the same eigenvalue. (In this case the eigenvalue is said to be “degenerate”, which sounds like an aspersion cast upon the morals of the eigenvalue but which is really just poor choice of terminology again.) Show that any vector of the form $c_1\mathbf{x} + c_2\mathbf{y}$ is an eigenvector with the same eigenvalue.

H.8 How to diagonalize a symmetric matrix

We saw in section H.3 that for any 2×2 symmetric matrix, represented in its initial basis by, say,

$$\begin{pmatrix} a & b \\ b & c \end{pmatrix}, \quad (10.8.39)$$

a simple rotation of axes would produce a new coordinate system in which the matrix representation is diagonal:

$$\begin{pmatrix} d_1 & 0 \\ 0 & d_2 \end{pmatrix}. \quad (10.8.40)$$

These two matrices are related through

$$\begin{pmatrix} d_1 & 0 \\ 0 & d_2 \end{pmatrix} = R(\phi) \begin{pmatrix} a & b \\ b & c \end{pmatrix} R^{-1}(\phi), \quad (10.8.41)$$

where $R(\phi)$ is the rotation matrix (H.3). Problem H.10 gave a direct way to find the desired rotation. However this direct technique is cumbersome and doesn’t generalize readily to higher dimensions. This section presents a different technique, which relies on eigenvalues and eigenvectors, that is more efficient and that generalizes readily to complex-valued matrices and to matrices in any dimension, but that is somewhat sneaky and conceptually roundabout.

We begin by noting that any vector lying along the x' -axis (of the preferred coordinate system) is an eigenvector. For example, the vector $5\hat{i}'$ is represented (in the preferred coordinate system) by

$$\begin{pmatrix} 5 \\ 0 \end{pmatrix}. \quad (10.8.42)$$

Multiplying this vector by the matrix in question gives

$$\begin{pmatrix} d_1 & 0 \\ 0 & d_2 \end{pmatrix} \begin{pmatrix} 5 \\ 0 \end{pmatrix} = d_1 \begin{pmatrix} 5 \\ 0 \end{pmatrix}, \quad (10.8.43)$$

so $5\hat{i}'$ is an eigenvector with eigenvalue d_1 . The same holds for any scalar multiple of \hat{i}' , whether positive or negative. Similarly, any scalar multiple of \hat{j}' is an eigenvector with eigenvalue d_2 . In short, the two elements on the diagonal in the preferred (diagonal) representation are the two eigenvalues, and the two unit vectors \hat{i}' and \hat{j}' of the preferred coordinate system are two of the eigenvectors. Thus finding the eigenvectors and eigenvalues of a matrix gives you the information needed to diagonalize that matrix. The unit vectors \hat{i}' and \hat{j}' constitute an “orthonormal basis of eigenvectors”. The eigenvectors even give the rotation matrix directly, as described in the next paragraph.

Let’s call the rotation matrix

$$B = \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} \quad (10.8.44)$$

so that the inverse (transpose) matrix is

$$B^{-1} = B^\dagger = \begin{pmatrix} b_{11} & b_{21} \\ b_{12} & b_{22} \end{pmatrix}. \quad (10.8.45)$$

The representation of \hat{i}' in the preferred basis is

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad (10.8.46)$$

so its representation in the initial basis is (see equation H.2)

$$B^\dagger \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} b_{11} & b_{21} \\ b_{12} & b_{22} \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} b_{11} \\ b_{12} \end{pmatrix}. \quad (10.8.47)$$

Similarly, the representation of \hat{j}' in the initial basis is

$$B^\dagger \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} b_{11} & b_{21} \\ b_{12} & b_{22} \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} b_{21} \\ b_{22} \end{pmatrix}. \quad (10.8.48)$$

Thus the rotation matrix is

$$B = \begin{pmatrix} \text{initial rep. of } \hat{i}', \text{ on its side} \\ \text{initial rep. of } \hat{j}', \text{ on its side} \end{pmatrix}. \quad (10.8.49)$$

Example

Suppose we need to find a diagonal representation for the matrix

$$T = \begin{pmatrix} 7 & 3 \\ 3 & 7 \end{pmatrix}. \quad (10.8.50)$$

First we search for the special vectors—the eigenvectors—such that

$$\begin{pmatrix} 7 & 3 \\ 3 & 7 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \lambda \begin{pmatrix} x \\ y \end{pmatrix}. \quad (10.8.51)$$

At the moment, we don't know either the eigenvalue λ or the associated eigenvector (x, y) . Thus it seems that (bad news) we are trying to solve two equations for three unknowns:

$$\begin{aligned} 7x + 3y &= \lambda x \\ 3x + 7y &= \lambda y \end{aligned} \quad (10.8.52)$$

Remember, however, that there is not one *single* eigenvector: any multiple of an eigenvector is also an eigenvector. (Alternatively, any vector on the line that extends the eigenvector is another eigenvector.) We only need one of these eigenvectors, so let's take the one that has $x = 1$ (i.e. the vector on the extension line where it intersects the vertical line $x = 1$). (This technique will fail if we have the bad luck that our actual eigenvector is vertical and hence never passes through the line $x = 1$.) So we really have two equations in two unknowns:

$$\begin{aligned} 7 + 3y &= \lambda \\ 3 + 7y &= \lambda y \end{aligned}$$

but note that they are not linear equations. . . the damnable product λy in the lower right corner means that all our techniques for solving linear equations go right out the window. We *can* solve these two equations for λ and y , but there's an easier, if somewhat roundabout, approach.

Finding eigenvalues

Let's go back to equation (H.52) and write it as

$$\begin{pmatrix} 7 & 3 \\ 3 & 7 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} - \lambda \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \quad (10.8.53)$$

Then

$$\begin{pmatrix} 7 & 3 \\ 3 & 7 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} - \lambda \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad (10.8.54)$$

or

$$\begin{pmatrix} 7 - \lambda & 3 \\ 3 & 7 - \lambda \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \quad (10.8.55)$$

Let's think about this. It says that for some matrix $M = T - \lambda I$, we have

$$M \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \quad (10.8.56)$$

You know right away one vector (x, y) that satisfies this equation, namely $(x, y) = (0, 0)$. And most of the time, this is the only vector that satisfies the equation, because

$$\begin{pmatrix} x \\ y \end{pmatrix} = M^{-1} \begin{pmatrix} 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \quad (10.8.57)$$

We appear to have reached a dead end. The solution is $(x, y) = (0, 0)$, but the zero vector is not, by definition, considered an eigenvector of any transformation. (Because it always gives eigenvalue zero for any transformation.)

However, if the matrix M is *not invertible*, then there will be other solutions to

$$M \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \quad (10.8.58)$$

in addition to the trivial solution $(x, y) = (0, 0)$. Thus we must look for those special values of λ such that the so-called *characteristic matrix* M is not invertible. These values come if and only if the determinant of M vanishes. For this example, we have to find values of λ such that

$$\det \begin{pmatrix} 7-\lambda & 3 \\ 3 & 7-\lambda \end{pmatrix} = 0. \quad (10.8.59)$$

This is a quadratic equation in λ

$$(7-\lambda)^2 - 3^2 = 0 \quad (10.8.60)$$

called the characteristic equation. Its two solutions are

$$7 - \lambda = \pm 3 \quad (10.8.61)$$

or

$$\lambda = 7 \pm 3 = 10 \text{ or } 4 \quad (10.8.62)$$

We have found the two eigenvalues of our matrix!

Finding eigenvectors

Let's look now for the eigenvector associated with $\lambda = 4$. Equation (H.53)

$$7x + 3y = \lambda x$$

$$3x + 7y = \lambda y$$

still holds, but no longer does it look like two equations in three unknowns, because we are now interested in the case $\lambda = 4$:

$$7x + 3y = 4x$$

$$3x + 7y = 4y$$

Following our nose gives

$$3x + 3y = 0$$

$$3x + 3y = 0$$

and when we see this our heart skips a beat or two. . . a degenerate system of equations! Relax and rest your heart. This system has an infinite number of solutions and it's supposed to have an infinite number of solutions, because any multiple of an eigenvector is also an eigenvector. The eigenvectors associated with $\lambda = 4$ are any multiple of

$$\begin{pmatrix} 1 \\ -1 \end{pmatrix}. \quad (10.8.63)$$

An entirely analogous search for the eigenvectors associated with $\lambda = 10$ finds any multiple of

$$\begin{pmatrix} 1 \\ 1 \end{pmatrix}. \quad (10.8.64)$$

Tidying up

We have the two sets of eigenvectors, but which shall we call $\hat{\mathbf{i}}'$ and which $\hat{\mathbf{j}}'$? This is a matter of individual choice, but my choice is usually to make the transformation be a rotation (without reflection) through a small positive angle. Our new, preferred coordinate system is related to the original coordinates by a simple rotation of 45° if we choose

$$\hat{\mathbf{i}}' = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad \text{and} \quad \hat{\mathbf{j}}' = \frac{1}{\sqrt{2}} \begin{pmatrix} -1 \\ 1 \end{pmatrix}. \quad (10.8.65)$$

(Note that we have also “normalized the basis”, i.e. selected the basis vectors to have magnitude unity.) Given this choice, the orthogonal rotation matrix that changes coordinates from the original to the preferred system is (see equation H.50)

$$\mathbf{B} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} \quad (10.8.66)$$

and the diagonalized matrix (or, more properly, the representation of the matrix in the preferred coordinate system) is

$$\begin{pmatrix} 10 & 0 \\ 0 & 4 \end{pmatrix}. \quad (10.8.67)$$

You don't believe me? Then multiply out

$$\mathbf{B} \begin{pmatrix} 7 & 3 \\ 3 & 7 \end{pmatrix} \mathbf{B}^\dagger \quad (10.8.68)$$

and see for yourself.

Problems

*H.14 Problem: Diagonalize a 2×2 matrix**

Diagonalize the matrix

$$\begin{pmatrix} 26 & 12 \\ 12 & 19 \end{pmatrix}. \quad (10.8.69)$$

- Find its eigenvalues.
- Find its eigenvectors, and verify that they are orthogonal.
- Sketch the eigenvectors, and determine the signs and sequence most convenient for assigning axes. (That is, should the first eigenvector you found be called $\hat{\mathbf{i}}'$, $-\hat{\mathbf{i}}'$, or $\hat{\mathbf{j}}'$?)
- Find the matrix that translates from the initial basis to the basis of eigenvectors produced in part (c.).
- Verify that the matrix produced in part (d.) is orthogonal.
- Verify that the representation of the matrix above in the basis of eigenvectors is diagonal.
- (Optional.) What is the rotation angle?

H.15 Problem: Eigenvalues of a 2×2 matrix

Show that the eigenvalues of

$$\begin{pmatrix} a & b \\ b & c \end{pmatrix} \quad (10.8.70)$$

are

$$\lambda = \frac{1}{2} \left[(a+c) \pm \sqrt{(a-c)^2 + 4b^2} \right] \quad (10.8.71)$$

Under what circumstances is an eigenvalue complex valued? Under what circumstances are the two eigenvalues the same?

H.16 Problem: Diagonalize a 3×3 matrix

Diagonalize the matrix

$$\frac{1}{625} \begin{pmatrix} 1182 & -924 & 540 \\ -924 & 643 & 720 \\ 540 & 720 & -575 \end{pmatrix} \quad (10.8.72)$$

a. Find its eigenvalues by showing that the characteristic equation is

$$\lambda^3 - 2\lambda^2 - 5\lambda + 6 = (\lambda - 3)(\lambda + 2)(\lambda - 1) = 0 \quad (10.8.73)$$

b. Find its eigenvectors, and verify that they are orthogonal.

c. Show that the translation matrix can be chosen to be

$$B = \frac{1}{25} \begin{pmatrix} 20 & -15 & 0 \\ 9 & 12 & -20 \\ 12 & 16 & 15 \end{pmatrix} \quad (10.8.74)$$

Why did I use the phrase “the translation matrix can be chosen to be” rather than “the translation matrix is”?

H.17 Problem: A 3×3 matrix eigenproblem

Find the eigenvalues and associated eigenvectors for the matrix

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \\ 3 & 4 & 5 \end{pmatrix}. \quad (10.8.75)$$

H.9 A glance at computer algorithms

Anyone who has worked even one of the problems in section H.8 knows that diagonalizing a matrix is no picnic: there’s a lot of mundane arithmetic involved and it’s very easy to make mistakes. This is a problem ripe for computer solution. One’s first thought is to program a computer to solve the problem using the same technique that we used to solve it on paper: first find the eigenvalues through the characteristic equation, then find the eigenvectors through a degenerate set of linear equations.

This turns out to be a very poor algorithm for automatic computation. The effective algorithm is to choose a matrix B such that the off-diagonal elements of

$$BAB^{-1} \quad (10.8.76)$$

are smaller than the off-diagonal elements of A . Then choose another, and another. Go through this process again and again until the off-diagonal elements have been ground down to machine zero. There are many strategies for choosing the series of B matrices. These are well-described in any edition of *Numerical Recipes*.⁴

When you need to diagonalize matrices numerically, I urge you to look at *Numerical Recipes* to see what’s going on, but I urge you *not* to code these algorithms yourself. These algorithms rely in an essential way on the fact that computer arithmetic is approximate rather than exact, and hence they are quite tricky to implement. Instead of coding the algorithms yourself, I recommend that you use the implementations in either LAPACK⁵ (the Linear Algebra PACKage) or EISPACK.⁶ These packages are probably the finest computer software ever written, and they are free. They can be obtained through the “Guide to Available Mathematical Software” (GAMS) at <http://gams.nist.gov>.

H.10 A glance at non-symmetric matrices and the Jordan form

Many of the matrices that arise in applications are symmetric and hence the results of the previous sections are the only ones needed. But every once in a while you do encounter a non-symmetric matrix and this section gives you a guide to treating them. It is just an introduction and treats only 2×2 matrices.

Given a non-symmetric matrix, the first thing to do is rotate the axes to make the matrix representation triangular, as discussed in problem H.12:

$$\begin{pmatrix} a & b \\ 0 & c \end{pmatrix}. \quad (10.8.77)$$

Note that $b \neq 0$ because otherwise the matrix would be symmetric and we would already be done. In this case vectors on the x -axis are eigenvectors because

$$\begin{pmatrix} a & b \\ 0 & c \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = a \begin{pmatrix} 1 \\ 0 \end{pmatrix}. \quad (10.8.78)$$

Are there any other eigenvectors? The equation

$$\begin{pmatrix} a & b \\ 0 & c \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = a \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad (10.8.79)$$

tells us that

$$\begin{aligned} ax + by &= \lambda x \\ cy &= \lambda y \end{aligned}$$

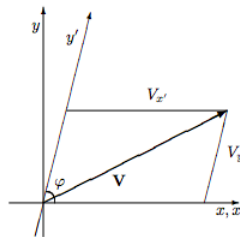
whence $\lambda \neq c$ and the eigenvector has polar angle θ where

$$\tan \theta = \frac{c - a}{b}. \quad (10.8.80)$$

Note that if $c = a$ (the “degenerate” case: both eigenvalues are the same) then $\theta = 0$ or $\theta = \pi$. In this case all of the eigenvectors are on the x -axis.

Diagonal form

We already know that that a rotation of orthogonal (Cartesian) coordinates will not diagonalize this matrix. We must instead transform to a skew coordinate system in which the axes are not perpendicular.



Note that in with oblique axes, the coordinates are given by

$$\mathbf{V} = V_{x'} \hat{\mathbf{i}}' + V_{y'} \hat{\mathbf{j}}' \quad (10.8.81)$$

but, because $\hat{\mathbf{i}}'$ and $\hat{\mathbf{j}}'$ are not perpendicular, it is *not true* that

$$V_{x'} = \mathbf{V} \cdot \hat{\mathbf{i}}'. \quad \text{NO!} \quad (10.8.82)$$

A little bit of geometry will convince you that the name of the vector \mathbf{V} changes according to

$$\begin{pmatrix} V_{x'} \\ V_{y'} \end{pmatrix} = B \begin{pmatrix} V_x \\ V_y \end{pmatrix}, \quad (10.8.83)$$

where

$$B = \frac{1}{\sin \varphi} \begin{pmatrix} \sin \varphi & -\cos \varphi \\ 0 & 1 \end{pmatrix}. \quad (10.8.84)$$

This matrix is *not* orthogonal. In fact its inverse is

$$B^{-1} = \begin{pmatrix} 1 & \cos \varphi \\ 0 & \sin \varphi \end{pmatrix}. \quad (10.8.85)$$

Finally, note that we cannot have $\varphi = 0$ or $\varphi = \pi$, because then both $V_{x'}$ and $V_{y'}$ would give information about the horizontal component of the vector, and there would be no information about the vertical component of the vector.

What does this say about the representations of tensors (or, equivalently, of linear transformations)? The “name translation” argument of equation (H.27) still applies, so

$$\mathbf{T}' = \mathbf{B}\mathbf{T}\mathbf{B}^{-1}. \quad (10.8.86)$$

Using the explicit matrices already given, this says

$$\mathbf{T}' = \frac{1}{\sin \varphi} \begin{pmatrix} \sin \varphi & -\cos \varphi \\ 0 & 1 \end{pmatrix} \begin{pmatrix} a & b \\ 0 & c \end{pmatrix} \begin{pmatrix} 1 & \cos \varphi \\ 0 & \sin \varphi \end{pmatrix} = \begin{pmatrix} a & (a-c)\cos \varphi + b\sin \varphi \\ 0 & c \end{pmatrix}. \quad (10.8.87)$$

To make this diagonal, we need only choose a skew coordinate system where the angle φ gives

$$(a-c)\cos \varphi + b\sin \varphi = 0, \quad (10.8.88)$$

that is, one with

$$\tan \varphi = \frac{c-a}{b}. \quad (10.8.89)$$

Comparison with equation (H.81) shows that this simply means that the skew coordinate system should have its axes pointing along two eigenvectors. We have once again found an intimate connection between diagonal representations and eigenvectors, a connection which is exploited fully in abstract mathematical treatments of matrix diagonalization.

Once again we can do an accounting of information. In the initial coordinate system, the four elements of the matrix contain four independent pieces of information. In the diagonalizing coordinate system, two of those pieces are explicit in the matrix, and two are implicit in the two axis rotation angles needed to implement the diagonalization.

This procedure works almost all the time. But, if $a = c$, then it would involve $\varphi = 0$ or $\varphi = \pi$, and we have already seen that this is not an acceptable change of coordinates.

Degenerate case

Suppose our matrix has equal eigenvalues, $a = c$, so that it reads

$$\begin{pmatrix} a & b \\ 0 & a \end{pmatrix}. \quad (10.8.90)$$

If $b = 0$, then the matrix is already diagonal. (Indeed, in this case all vectors are eigenvectors with eigenvalue a , and the linear transformation is simply multiplication of each vector by a).

But if $b \neq 0$, then, as we have seen, the only eigenvectors are on the x -axis, and it is impossible to make a basis of eigenvectors. Only one thing can be done to make the matrix representation simpler than it stands in equation (H.91), and that is a shift in the scale used to measure the y -axis.

For example, suppose that in the (x, y) coordinate system, the y -axis is calibrated in inches. We wish to switch to the (x', y') system in which the y' -axis is calibrated in feet. There is no change in axis orientation or in the x -axis. It is easy to see that the two sets of coordinates are related through

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1/12 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 12 \end{pmatrix} \begin{pmatrix} x' \\ y' \end{pmatrix} \quad (10.8.91)$$

This process is sometimes called a “stretching” or a “scaling” of the y -axis.

The transformation represented by matrix (H.91) in the initial coordinate system is represented in the new coordinate system by

$$\begin{pmatrix} 1 & 0 \\ 0 & 1/12 \end{pmatrix} \begin{pmatrix} a & b \\ 0 & a \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 12 \end{pmatrix} = \begin{pmatrix} a & 12b \\ 0 & a \end{pmatrix}. \quad (10.8.92)$$

The choice of what to do now is clear. Instead of scaling the y -axis by a factor of 12, we can scale it by a factor of $1/b$, and produce a new matrix representation of the form

$$\begin{pmatrix} a & 1 \\ 0 & a \end{pmatrix}. \quad (10.8.93)$$

Where is the information in this case? In the initial coordinate system, the four elements of the matrix contain four independent pieces of information. In the new coordinate system, two of those pieces are explicit in the matrix, one is implicit in the rotation angle needed to implement the initial triangularization, and one is implicit in the y-axis scale transformation.

The Jordan form

Remarkably, the situation discussed above for 2×2 matrices covers all the possible cases for $n \times n$ matrices. That is, in n dimensional space, the proper combination of rotations, skews, and stretches of coordinate axes will bring the matrix representation (the “name”) of any tensor or linear transformation into a form where every element is zero except on the diagonal and on the superdiagonal. The elements on the diagonal are eigenvalues, and each element on the superdiagonal is either zero or one: zero if the two adjacent eigenvalues differ, either zero or one if they are the same. The warning of problem H.12 applies here as well: The eigenvalues on the diagonal may well be complex valued, and the same applies for the elements of the new basis vectors.

References

¹For example, Kenneth Hoffman and Ray Kunze, *Linear Algebra*, second edition (Prentice-Hall, Englewood Cliffs, New Jersey, 1971).

²For example, Jerry Marion and Stephen Thorton, *Classical Dynamics of Particles and Systems*, fourth edition (Saunders College Publishing, Fort Worth, Texas, 1995) section 11.2.

³For example, Jerry Marion and Stephen Thorton, *Classical Dynamics of Particles and Systems*, fourth edition (Saunders College Publishing, Fort Worth, Texas, 1995) section 11.7.

⁴W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, *Numerical Recipes* (Cambridge University Press, Cambridge, U.K., 1992).

⁵E. Anderson, *et al.*, *LAPACK Users’ Guide* (SIAM, Philadelphia, 1992).

⁶B.T. Smith, *et al.*, *Matrix Eigensystem Routines—EISPACK Guide* (Springer-Verlag, Berlin, 1976).

¹Although all rotation matrices are orthogonal, there are orthogonal matrices that are not rotation matrices: see problem H.4.

²Or, to be absolutely precise, the particle located at the point represented by the vector with components (x, y) .

³An efficient algorithm for diagonalization is discussed in section H.8. For the moment, we are more interested in knowing that a diagonal matrix representation must exist than in knowing how to most easily find that preferred coordinate system.

10.8: H- Tutorial on Matrix Diagonalization is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.