

## 9

### *Eigenvalues and Eigenvectors*

Consider the matrix

$$\mathbf{A} = \begin{pmatrix} 2 & 7 \\ -1 & -6 \end{pmatrix}$$

Multiplying  $\mathbf{A}$  by the vector  $\mathbf{x}_1 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$  gives an interesting result.

$$\mathbf{A}\mathbf{x}_1 = \begin{pmatrix} 2 & 7 \\ -1 & -6 \end{pmatrix} \begin{pmatrix} -1 \\ 1 \end{pmatrix} = \begin{pmatrix} 5 \\ -5 \end{pmatrix} = -5 \begin{pmatrix} -1 \\ 1 \end{pmatrix} = -5\mathbf{x}_1$$

Similarly, with  $\mathbf{x}_2 = \begin{pmatrix} -7 \\ 1 \end{pmatrix}$ :

$$\mathbf{A}\mathbf{x}_2 = \begin{pmatrix} 2 & 7 \\ -1 & -6 \end{pmatrix} \begin{pmatrix} -7 \\ 1 \end{pmatrix} = \begin{pmatrix} -7 \\ 1 \end{pmatrix} = \mathbf{x}_2$$

In both cases, multiplication by  $\mathbf{A}$  returned a scalar multiple of the vector (-5 for  $\mathbf{x}_1$  and 1 for  $\mathbf{x}_2$ ). This is not a property of solely the matrix  $\mathbf{A}$ , since the vector  $\mathbf{x}_3 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$  is not transformed by a single scalar.

$$\mathbf{A}\mathbf{x}_3 = \begin{pmatrix} 2 & 7 \\ -1 & -6 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 9 \\ 5 \end{pmatrix} \neq \lambda\mathbf{x}_3$$

Similarly, the results we are seeing are not properties of the vectors  $\mathbf{x}_1$  and  $\mathbf{x}_2$ , since they do not become scalar multiples of themselves when multiplied by other matrices.

$$\mathbf{B} = \begin{pmatrix} 2 & 1 \\ -3 & 0 \end{pmatrix}$$

$$\mathbf{B}\mathbf{x}_1 = \begin{pmatrix} 2 & 1 \\ -3 & 0 \end{pmatrix} \begin{pmatrix} -1 \\ 1 \end{pmatrix} = \begin{pmatrix} -1 \\ 3 \end{pmatrix} \neq \lambda\mathbf{x}_1$$

$$\mathbf{B}\mathbf{x}_2 = \begin{pmatrix} 2 & 1 \\ -3 & 0 \end{pmatrix} \begin{pmatrix} -7 \\ 1 \end{pmatrix} = \begin{pmatrix} -13 \\ 21 \end{pmatrix} \neq \lambda\mathbf{x}_2$$

The phenomena we're observing is a result of the pairing between the matrix  $\mathbf{A}$  and the vectors  $\mathbf{x}_1$  and  $\mathbf{x}_2$ . In general, we see that multiplying a vector by a matrix returns a scalar multiple of the vector, or

$$\mathbf{Ax} = \lambda\mathbf{x}$$

Any vector  $\mathbf{x}$  that obeys the above relationship is called an *eigenvector* of the matrix  $\mathbf{A}$ . The scalar  $\lambda$  is called the *eigenvalue* associated with the eigenvector  $\mathbf{x}$ . The vector  $\mathbf{x}$  is an eigenvector of the matrix  $\mathbf{A}$ ; it is not generally an eigenvector of other matrices.

In the example above, the matrix  $\mathbf{A} = \begin{pmatrix} 2 & 7 \\ -1 & -6 \end{pmatrix}$  has two eigenvectors,  $\mathbf{v}_1 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$  with eigenvalue  $\lambda_1 = -5$ , and  $\mathbf{v}_2 = \begin{pmatrix} -7 \\ 1 \end{pmatrix}$  with eigenvalue  $\lambda_2 = 1$ .

### 9.1 Properties of Eigenvectors and Eigenvalues

Only square matrices have eigenvectors and eigenvalues. An  $n$  by  $n$  matrix of real numbers can have up to  $n$  distinct eigenvectors. Each eigenvector is associated with an eigenvalue, although the eigenvalues can be duplicated. Said another way, two eigenvectors  $\mathbf{v}_1$  and  $\mathbf{v}_2$  of a matrix will never be the same, but the corresponding eigenvalues  $\lambda_1$  and  $\lambda_2$  can be identical.

Although the number of eigenvectors may vary, all eigenvectors for a matrix are linearly independent. Thus, if an  $n$  by  $n$  matrix has  $n$  eigenvectors, these vectors can be used as a basis (called an *eigenbasis*). If an eigenbasis exists for a matrix, decomposing vectors over this basis simplifies the process of matrix multiplication. To illustrate, imagine we decompose the vector  $\mathbf{x}$  over a set of eigenvectors  $\mathbf{v}_1, \dots, \mathbf{v}_n$ . Decomposing  $\mathbf{x}$  means we can find coefficients  $a_1, \dots, a_n$  such that

$$\mathbf{x} = a_1\mathbf{v}_1 + \dots + a_n\mathbf{v}_n$$

Now let's compute the product  $\mathbf{Ax}$ . We multiply both sides of the decomposition by  $\mathbf{A}$ .

$$\mathbf{Ax} = \mathbf{A}(a_1\mathbf{v}_1 + \dots + a_n\mathbf{v}_n)$$

We distribute the matrix  $\mathbf{A}$  into the sum on the right hand side and note that the constants  $a_i$  can be moved in front of the matrix multiplication.

$$\mathbf{Ax} = a_1\mathbf{A}\mathbf{v}_1 + \dots + a_n\mathbf{A}\mathbf{v}_n$$

Remember that  $\mathbf{v}_1, \dots, \mathbf{v}_n$  are eigenvectors of  $\mathbf{A}$ , so  $\mathbf{A}\mathbf{v}_i = \lambda_i\mathbf{v}_i$ . We can simplify the previous expression to

$$\mathbf{Ax} = a_1\lambda_1\mathbf{v}_1 + \dots + a_n\lambda_n\mathbf{v}_n$$

Eigenvectors were originally called *characteristic* vectors, as they describe the character of the matrix. German mathematicians dropped this nomenclature in favor of the German prefix "eigen-", which mean "own". An eigenvector can be viewed as one of a matrix's "own" vectors since it is not rotated when transformed by multiplication.

To understand why the matrix must be square, remember that a non-square matrix with  $m$  rows and  $n$  columns transforms an  $n$ -dimensional vectors into an  $m$ -dimensional vector. Clearly, the  $m$ -dimensional output cannot be the  $n$ -dimensional input multiplied by a scalar!

An  $n$  by  $n$  matrix with  $n$  eigenvectors and  $n$  distinct eigenvalues is called a *perfect matrix*. As the name implies, perfect matrices are great to find, but somewhat uncommon.

We don't need to perform the multiplication at all! Instead, we can scale each eigenvector by the eigenvalue. Multiplying again by the matrix  $\mathbf{A}$  multiplies each eigenvector by its eigenvalue.

$$\begin{aligned}\mathbf{A}^2 \mathbf{x} &= a_1 \lambda_1^2 \mathbf{v}_1 + \cdots + a_n \lambda_n^2 \mathbf{v}_n \\ \mathbf{A}^k \mathbf{x} &= a_1 \lambda_1^k \mathbf{v}_1 + \cdots + a_n \lambda_n^k \mathbf{v}_n\end{aligned}$$

We use the notation  $\mathbf{A}^2$  to denote  $\mathbf{A}\mathbf{A}$ ,  $\mathbf{A}^3$  for  $\mathbf{A}\mathbf{A}\mathbf{A}$ , and  $\mathbf{A}^k$  for the product of  $k$  matrices  $\mathbf{A}$ .

## 9.2 Computing Eigenvectors and Eigenvalues

We can use the relationship between matrix multiplication and eigenvalues to find eigenvectors for any matrix. Our computational approach is based on the following theorem.

**Theorem.** *Given any (random) vector  $\mathbf{b}$ , repeated multiplication by the matrix  $\mathbf{A}$  will converge to the eigenvector of  $\mathbf{A}$  with the largest magnitude eigenvalue – provided the largest eigenvalue is unique. Said another way,*

$$\lim_{k \rightarrow \infty} \mathbf{A}^k \mathbf{b} = \mathbf{v}_{\max}$$

*Proof.* We know that the product  $\mathbf{A}\mathbf{x}$  can be expressed as a linear combination of the eigenvectors and eigenvalues of  $\mathbf{A}$ , i.e.  $\mathbf{A}\mathbf{x} = a_1 \lambda_1 \mathbf{v}_1 + \cdots + a_n \lambda_n \mathbf{v}_n$ . Thus

$$\lim_{k \rightarrow \infty} \mathbf{A}^k \mathbf{b} = \lim_{k \rightarrow \infty} (a_1 \lambda_1^k \mathbf{v}_1 + \cdots + a_n \lambda_n^k \mathbf{v}_n)$$

As  $k$  increases, the values  $\lambda_i^k$  grow very large. However, the  $\lambda_i$  do not grow at the same rate. The largest eigenvalue will grow the fastest. At very large values of  $k$ , the term associated with the largest eigenvalue will dominate the entire sum, so the result will point in only the direction of the associated eigenvector. Note that convergence to a single eigenvector requires that the largest eigenvalue be distinct. If two eigenvectors have the same (largest) eigenvalue, both terms in the above sum would “blow up” at the same rate. Repeated multiplications by  $\mathbf{A}$  would then converge to the sum of the two associated eigenvectors.  $\square$

The above theorem allows us to find the eigenvector paired with the largest eigenvalue. While the direction of the eigenvector doesn't change, its magnitude grows as the number of multiplication of  $\mathbf{A}$  increases. If convergence is slow, we might need to work with numbers before finding the eigenvector. To avoid numerical difficulties, we renormalize the vector after every multiplication by  $\mathbf{A}$ . This algorithm is called the Power Iteration method, which proceeds as follows:

1. Choose a random vector  $\mathbf{b}_0$ . For fastest convergence, it helps to choose a vector close to  $\mathbf{v}_{\max}$  if possible. Normalize this vector to product  $\hat{\mathbf{b}}_0 = \mathbf{b}_0 / \|\mathbf{b}_0\|$ .

2. Compute vector  $\mathbf{b}_1 = \mathbf{A}\hat{\mathbf{b}}_0$ . Normalize this vector to give  $\hat{\mathbf{b}}_1 = \mathbf{b}_1 / \|\mathbf{b}_1\|$ .
3. Repeat step 2 to product  $\hat{\mathbf{b}}_2, \hat{\mathbf{b}}_3, \dots, \hat{\mathbf{b}}_k$ . Stop when all entries of  $\hat{\mathbf{b}}_k$  do not change from the entries in  $\hat{\mathbf{b}}_{k-1}$ . The vector  $\hat{\mathbf{b}}_k$  is an eigenvector of  $\mathbf{A}$ .

Now that we have the eigenvector  $\mathbf{v}_{\max}$ , how do we find the associated eigenvalue  $\lambda_{\max}$ ? We know that  $\mathbf{v}_{\max}$  is an eigenvector of  $\mathbf{A}$ , to  $\mathbf{A}\mathbf{v}_{\max} = \lambda_{\max}\mathbf{v}_{\max}$ . The  $i$ th element in  $\mathbf{A}\mathbf{v}_{\max}$  should be equal to  $\lambda_{\max}$  times the  $i$ th element in  $\mathbf{v}_{\max}$ . However, since we only found a numerical approximation to the  $\mathbf{v}_{\max}$ , the estimate for  $\lambda_{\max}$  from each element in  $\mathbf{v}_{\max}$  might differ slightly. To “smooth out” these variations, compute the eigenvalue using the Rayleigh quotient:

$$\lambda_{\max} = \frac{\mathbf{v}_{\max} \cdot \mathbf{A}\mathbf{v}_{\max}}{\mathbf{v}_{\max} \cdot \mathbf{v}_{\max}}$$

The dot product in the Rayleigh quotient averages out all of the small discrepancies between our estimate  $\mathbf{v}_{\max}$  and the true largest eigenvector. The Rayleigh quotient provides a numerically stable estimate of the largest eigenvalue.

Now that we’ve found the first eigenvector, how do we find the others? If we start the Power Iteration method over again using the matrix  $(\mathbf{A} - \lambda_{\max}\mathbf{I})$  instead of  $\mathbf{A}$ , the algorithm will converge to the eigenvector associated with the second largest eigenvalue. We can subtract this eigenvalue from  $\mathbf{A}$  and repeat to find the third eigenvector, and so on. Proving Power Iteration is able to find subsequent eigenvectors is beyond the scope of this course. However, as we’ll see later, finding only the first eigenvector is sufficient for addressing a number of interesting problems.

### 9.2.1 Eigenvalues and Eigenvectors in MATLAB

The MATLAB function `eig` computes eigenvalues and eigenvectors. The statement `[V,L] = eig(A)` involving an  $n$  by  $n$  matrix  $\mathbf{A}$  returns two  $n$  by  $n$  matrices:

- Each column of the matrix  $\mathbf{V}$  is an eigenvector  $\mathbf{A}$ .
- The matrix  $\mathbf{L}$  is a diagonal matrix. The  $i$ th entry on the diagonal is the eigenvalue associated with the  $i$ th column in  $\mathbf{V}$ .

Remember that any vector that points in the same direction as an eigenvector of a matrix is also an eigenvector of that matrix. If the eigenvectors returned by computational systems like MATLAB are not what you expect, remember that they may be normalized or scaled – but still point along the same direction.

The eigenvector associated with the largest magnitude eigenvalue is called the *leading eigenvector*.

To see why the Raleigh quotient works, consider an eigenvector  $\mathbf{v}$  for matrix  $\mathbf{A}$  with associated eigenvalue  $\lambda$ . Then

$$\frac{\mathbf{v} \cdot \mathbf{A}\mathbf{v}}{\mathbf{v} \cdot \mathbf{v}} = \frac{\mathbf{v} \cdot (\lambda\mathbf{v})}{\mathbf{v} \cdot \mathbf{v}} = \lambda \frac{\mathbf{v} \cdot \mathbf{v}}{\mathbf{v} \cdot \mathbf{v}} = \lambda$$

### 9.3 Applications

Eigenvalue and eigenvectors can be used to solve a number of interesting engineering and data science problems.

#### 9.3.1 Solving Systems of ODEs

Consider the linear system of ODEs

$$\begin{aligned}\frac{dx_1}{dt} &= x_1 + 2x_2 \\ \frac{dx_2}{dt} &= 3x_1 + 2x_2\end{aligned}$$

with initial conditions  $x_1(0) = 0$  and  $x_2(0) = -4$ . We can write this system using vectors and matrices as

$$\frac{d\mathbf{x}}{dt} = \mathbf{A}\mathbf{x}, \quad \mathbf{x}(0) = \mathbf{x}_0$$

where for the example above

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \quad \mathbf{A} = \begin{pmatrix} 1 & 2 \\ 3 & 2 \end{pmatrix}, \quad \mathbf{x}_0 = \begin{pmatrix} 0 \\ -4 \end{pmatrix}$$

If we know the eigenvectors  $\mathbf{v}_1, \dots, \mathbf{v}_n$  and eigenvalues  $\lambda_1, \dots, \lambda_n$  for the matrix  $\mathbf{A}$ , we can compute the solution as

$$\mathbf{x}(t) = c_1 \mathbf{v}_1 e^{\lambda_1 t} + c_2 \mathbf{v}_2 e^{\lambda_2 t} + \dots + c_n \mathbf{v}_n e^{\lambda_n t}$$

The scalars  $c_1, \dots, c_n$  are the constants of integration. To find these values, notice what happens to our solution at time  $t = 0$ :

$$\mathbf{x}(0) = \mathbf{x}_0 = c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2 + \dots + c_n \mathbf{v}_n$$

At  $t = 0$ , the right hand side is a decomposition of the initial conditions  $\mathbf{x}_0$ . If we collect the eigenvectors as columns of a matrix  $\mathbf{V} = (\mathbf{v}_1 \mathbf{v}_2 \dots \mathbf{v}_n)$ , we can find the constants  $c_1, \dots, c_n$  by solving the linear system

$$\mathbf{V} \begin{pmatrix} c_1 \\ \vdots \\ c_n \end{pmatrix} = \mathbf{x}_0$$

Returning to our original example, the matrix

$$\mathbf{A} = \begin{pmatrix} 1 & 2 \\ 3 & 2 \end{pmatrix}$$

has eigenvalue/eigenvector pairs

$$\lambda_1 = -1, \quad \mathbf{v}_1 = \begin{pmatrix} -1 \\ 1 \end{pmatrix} \quad \text{and} \quad \lambda_2 = 4, \quad \mathbf{v}_2 = \begin{pmatrix} 2 \\ 3 \end{pmatrix}$$

This solution requires the matrix  $\mathbf{A}$  be perfect and therefore have a complete set of eigenvectors.

The function  $f(t) = e^{\lambda t}$  is an *eigenfunction* of the derivative operator, i.e.

$$\frac{d}{dt} f(t) = \lambda e^{\lambda t} = \lambda f(t)$$

The solution of a system of linear ODEs is the product of the eigenvectors of  $\mathbf{A}$  and the eigenfunctions of  $\frac{d\mathbf{x}}{dt}$ .

The integration constants  $c_1$  and  $c_2$  are defined by the system  $\mathbf{V}\mathbf{c} = \mathbf{x}_0$ , which for this example is

$$\begin{pmatrix} -1 & 2 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} = \begin{pmatrix} 0 \\ -4 \end{pmatrix}$$

Solving the above equations reveals  $c_1 = -4/5$  and  $c_2 = -8/5$ . The final solution to this systems of ODEs is

$$\mathbf{x}(t) = -\frac{8}{5} \begin{pmatrix} -1 \\ 1 \end{pmatrix} e^{-t} - \frac{4}{5} \begin{pmatrix} 2 \\ 3 \end{pmatrix} e^{4t}$$

### 9.3.2 Stability of Linear ODEs

The eigenvalues of  $\mathbf{A}$  are sufficient to tell if the system  $\frac{d\mathbf{x}}{dt} = \mathbf{A}\mathbf{x}$  is stable. For a system of linear ODEs to be stable, all eigenvalues of  $\mathbf{A}$  must be nonpositive. If the eigenvalues are all negative, each term  $e^{\lambda_i t}$  goes to zero at long times, so all variables in the system to go zero. If any of the eigenvalues are zero, the system is still stable (provided all other eigenvalues are negative), but the system will go to a constant value  $c_i \mathbf{v}_i$ , where  $\mathbf{v}_i$  is the eigenvector associated with the zero eigenvalue.

### 9.3.3 Positive Definite Matrices

A symmetric matrix  $\mathbf{A}$  is *positive definite* (p.d.) if  $\mathbf{x}^T \mathbf{A} \mathbf{x} > 0$  for all nonzero vectors  $\mathbf{x}$ . If a matrix  $\mathbf{A}$  satisfies the slightly relaxed requirement that  $\mathbf{x}^T \mathbf{A} \mathbf{x} \geq 0$  for all nonzero  $\mathbf{x}$ , we say that  $\mathbf{A}$  is *positive semi-definite* (p.s.d.).

Knowing that a matrix is positive (semi-)definite is useful for quadratic programming problems like the Support Vector Machine classifier. The quadratic function  $f(\mathbf{x}) = \mathbf{x}^T \mathbf{Q} \mathbf{x}$  if and only if the matrix  $\mathbf{Q}$  is positive semi-definite. For optimization problems like quadratic programs, the convexity of the objective function has enormous implications. Convex quadratic programs must only have global optima, making them easy to solve using numerical algorithms.

Determining if a matrix is positive (semi-)definite can be difficult unless we use eigenvalues. Any matrix with only positive eigenvalues is positive definite, and any matrix with only nonnegative eigenvalues is positive semi-definite. For example, consider the  $2 \times 2$  identity matrix

$$\mathbf{I} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

The product  $\mathbf{x}^T \mathbf{I} \mathbf{x}$  is

$$\begin{pmatrix} x_1 & x_2 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = x_1^2 + x_2^2$$

Remember that a matrix  $\mathbf{A}$  is symmetric if  $\mathbf{A} = \mathbf{A}^T$ .

If  $\mathbf{Q}$  is positive definite (rather than just positive semi-definite) then  $\mathbf{x}^T \mathbf{Q} \mathbf{x}$  is strictly convex.

Since  $x_1^2 + x_2^2$  is greater than zero for all nonzero inputs  $x_1$  and  $x_2$ , the matrix  $\mathbf{I}$  is positive definite and all its eigenvalues should be positive. Indeed, the eigenvalues for the identity matrix are  $\lambda_1 = \lambda_2 = 1$ .

As another example, consider the matrix

$$\mathbf{A} = \begin{pmatrix} 1 & -2 \\ -2 & 1 \end{pmatrix}$$

The product  $\mathbf{x}^T \mathbf{A} \mathbf{x} = x_1^2 - 4x_1x_2 + x_2^2$ , which is not always positive. When  $x_1 = x_2 = 1$ , we see that  $x_1^2 - 4x_1x_2 + x_2^2 = -2$ . We know that  $\mathbf{A}$  is not positive definite (or even positive semi-definite), so  $\mathbf{A}$  should have at least one negative eigenvalue. As expected, the eigenvalues for  $\mathbf{A}$  are  $\lambda_1 = 3$  and  $\lambda_2 = -1$ .