Matrix Math for Data Science

Part 3 - Eigenvalues and EigenVectors
(Greg Simpson 2019, January)

1 Introduction and Rationale

This is the 3rd article in my matrix math series. I introduced some terms and covered matrix addition and subtraction in the first article. And I covered matrix multiplication in the second article. Each article assumes that you know why you want to add, subtract or multiply matrixes so they only talk about the mechanics of how you go about it. This article will be no different, except that it will focus on the mechanics of eigenvalues and eigenvectors. There is a lot of information out there to explain whay you may need the "Eigen's", so I do not want to duplicate that. I list a few links in the reference section at the end of this article.

Please feel free to leave feedback.

2 Definitions

We will start with a few definitions related to matrices. It helps if you think of them like database tables, so I will use database table imagry in my explanations.

columns the fields that define a table; they go left to right, horizontally

$$
\text{example of columns} \quad {}_{rows}\!\!\begin{array}{c}\text{columns} \\[-2pt]\end{array}\ \begin{array}{cccc}col_1 & col_2 & ... & col_n\end{array} \left[\begin{array}{cccc} \vdots & \vdots & \ddots & \vdots \end{array}\right]
$$

dimension numeric description of the number of rows and columns; in math problems, they usually say an "m by n" matrix or "m x n"

where m = rows and n = columns

$$
\text{3x3 : m=3, n=3} \qquad \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}
$$

$$
\text{2x3 : rows=2, cols=3} \qquad \begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix}
$$

$$
\text{2x4 : m=2, n=4} \qquad \begin{bmatrix} a & b & c & d \\ e & f & g & h \end{bmatrix}
$$

matrix an array that has entries in two directions (or more): an "m by n" or "m x n" array

(see the examples for 'dimension')

multiplication multiplier x multiplicand = product

reduced row eschelon form (RREF) see reference link below

rows the data that populates the table; each row usually has an entry for each column

$$
\text{example of rows} \quad \begin{array}{c} \\ row_1 \\ row_2 \\ \\ row_m \end{array} \begin{matrix} col_1 & col_2 & \ldots & col_n \\ \begin{bmatrix} \vdots & \vdots & \ldots & \vdots \\ \vdots & \vdots & \ldots & \vdots \\ \vdots & \vdots & \ldots & \vdots \end{bmatrix} \end{matrix}
$$

shape another term for dimension

vector an array that has entries in one direction:
1 row by n columns
or m rows by 1 column

$$
\text{row vector} \quad 1row \quad \begin{matrix} col_1 & col_2 & \ldots & col_n \\ \begin{bmatrix} a & b & \ldots & d \end{bmatrix} \end{matrix}
$$

$$
\text{column vector} \quad \begin{array}{c} \\ row_1 \\ row_2 \\ \\ row_m \end{array} \begin{matrix} col_1 \\ \begin{bmatrix} a \\ b \\ \vdots \end{bmatrix} \end{matrix}
$$

3 Matrix Eigenvalue Rules

For any lambda that satisfies

$$A \times \vec{v} = \lambda \times \vec{v}$$

for any Matrix(A), non-zero vector $\vec{v}$ and the Identity matrix $\mathbb{1}$

Then the determinant of ((lambda times the identity matrix) - A) must be equal to zero. The math formula that for that is :

$$\det(\lambda\mathbb{1} - A) = 0$$

Another way to say that is
"lambda is an eigenvalue of A if an only if that equation is true".

4   Matrix Eigenvalue Example

Let's look at an example of solving for Eigenvalues.
For matrix

$$A = \begin{bmatrix} 1 & 3 \\ 4 & 5 \end{bmatrix}$$

Given the formula from the rule above, solving this equation will find the eigenvalues of A

$$\det(\lambda \times \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 1 & 3 \\ 4 & 5 \end{bmatrix}) = 0$$

That becomes

$$\det(\begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} - \begin{bmatrix} 1 & 3 \\ 4 & 5 \end{bmatrix}) = 0$$

Then this

$$\det(\begin{bmatrix} \lambda - 1 & 0 - 3 \\ 0 - 4 & \lambda - 5 \end{bmatrix}) = 0$$

Taking the determinant is solved by :

$$((\lambda - 1) \times (\lambda - 5) - (-4 \times -3) = 0$$

Expanding that equation give us :

$$\lambda^2 - 5\lambda - 1\lambda + 5 - 12 = 0$$

or

$$\lambda^2 - 6\lambda - 7 = 0$$

Factoring that gives us (finally) :

$$(\lambda - 7) \times (\lambda + 1) = 0$$

And that means that the non-zero Eigenvalues for matrix A, are :

$$\lambda = 7; \lambda = -1$$

5 Matrix Eigenvector Rules

Eigenvectors are the set of vectors that satisfy the equation that we used for the determinant to find the eigenvalues.

In other words, you find eigenvalues using this equation:

$$\det(\lambda \mathbb{1} - A) = 0$$

And you find eigenvectors, by using each eigenvalue in place of lambda to find the NullSpace given by this equation:

$$N(\lambda \mathbb{1} - A)$$

The Nullspace of a matrix equals the Nullspace of the Reduced Row Eschelon Form (RREF) of the matrix.
The Eigenspace for each eigenvalue of a matrix is the set of eigenvectors that correspond to that eigenvalue.

So the algorithm to find the eigenvectors is as follows.

1. set **lambda** equal to each eigenvalue (a FOR loop)

2. solve the equation $(\lambda \mathbb{1} - A)$

3. find the Nullspace from step 2

   (a) reduce the matrix to RREF
   (b) solve for that times some vector $\vec{v} =$ the zero vector

.

    We will look at an example next. It should make things much more clear.

6   Matrix Eigenvector Example

We will find the eigenvectors that correspond to each eigenvalue from our previous example. These are also called the eigenspaces for our matrix A.

Our two eigenvalues from above were $\lambda = (-1)$ and $\lambda = 7$.
So we know we have 2 Nullspaces to find (one for each eigenvalue : -1 and 7).

6.1   Let $\lambda = (-1)$

Find, E $_{(-1)} = N(\lambda \mathbb{1} - A)$

Filling in our λ and matrix example values gives us

1. $N(\begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} - \begin{bmatrix} 1 & 3 \\ 4 & 5 \end{bmatrix})$

2. $= N(\begin{bmatrix} -2 & -3 \\ -4 & -6 \end{bmatrix})$

   (a) RREF R$_{(2,1)} - 2 = \begin{bmatrix} -2 & -3 \\ 0 & 0 \end{bmatrix}$

   (b) RREF R$_{(1)} \frac{-1}{2} = \begin{bmatrix} 1 & \frac{3}{2} \\ 0 & 0 \end{bmatrix}$

3. $= \begin{bmatrix} 1 & \frac{3}{2} \\ 0 & 0 \end{bmatrix} \times \begin{bmatrix} x \\ y \end{bmatrix} = \vec{0}$

   (a) $x + \frac{3}{2}y = 0$
   (b) $x = \frac{-3}{2}y$

4. $\begin{bmatrix} \frac{-3}{2}y \\ y \end{bmatrix} = \vec{0}$

5. Let y=1, then the eigenvector that corresponds to eigenvalue (-1) is

$$E_{(-1)} = \begin{bmatrix} \frac{-3}{2} \\ 1 \end{bmatrix}$$

6.2 Let $\lambda = (7)$

Find, $E_{(7)} = N(\lambda \mathbb{1} - A)$

Filling in our λ and matrix example values gives us

1. $N(\begin{bmatrix} 7 & 0 \\ 0 & 7 \end{bmatrix} - \begin{bmatrix} 1 & 3 \\ 4 & 5 \end{bmatrix})$

2. $= N(\begin{bmatrix} 6 & -3 \\ -4 & 2 \end{bmatrix})$

    (a) RREF $R_{(2,1)\frac{2}{3}} = \begin{bmatrix} 6 & -3 \\ 0 & 0 \end{bmatrix}$

    (b) RREF $R_{(1)\frac{1}{6}} = \begin{bmatrix} 1 & \frac{-1}{2} \\ 0 & 0 \end{bmatrix}$

3. $= \begin{bmatrix} 1 & \frac{-1}{2} \\ 0 & 0 \end{bmatrix} \times \begin{bmatrix} x \\ y \end{bmatrix} = \vec{0}$

    (a) $x - \frac{1}{2}y = 0$
    (b) $x = \frac{1}{2}y$

4. $\begin{bmatrix} \frac{1}{2}y \\ y \end{bmatrix} = \vec{0}$

5. Let y=1, then the eigenvector that corresponds to eigenvalue (7) is

$$E_{(7)} = \begin{bmatrix} \frac{1}{2} \\ 1 \end{bmatrix}$$

6.3 Eigenspace for our example

Eigenvector for $\lambda = -1 : \begin{bmatrix} -\frac{3}{2} \\ 1 \end{bmatrix}$

Eigenvector for $\lambda = +7 : \begin{bmatrix} \frac{1}{2} \\ 1 \end{bmatrix}$

7 Matrix Eigenvalue and Eigenvector example in python code

Here is a short and simple python program that implements our eigen examples.

7.0.1 EigenValue and EigenVector examples in python code

```python
# # matrix example
# A = [[1, 3],
#      [4, 5]]

# these code samples match the examples in the article

# example 1
print("Find the Eigenvalues for matrix A")

import numpy as np
from numpy import linalg as LA
A = np.array([[1,3],
              [4,5]])

w, v = LA.eig(A)
print(w)

'''
# example 2
print("Find the Eigenvectors for matrix A")

# eigenvectors do not match
#   see this post for an explanation
# https://stackoverflow.com/questions/31487264/numpy-eigenvalues-correct-but-eig

print(v)

u = v[:,0]
print(u)
print(np.dot(A,u))

u = v[:,1]
print(u)
print(np.dot(A,u))
'''
```

8 reference links

https://du.edu

https://en.wikipedia.org/wiki/Row_echelon_form

https://ritchieschool.du.edu/datascience/

https://www.geeksforgeeks.org/python-program-multiply-two-matrices/

https://www.khanacademy.org/math/linear-algebra/alternate-bases/eigen-everything/
v/linear-algebra-introduction-to-eigenvalues-and-eigenvectors

https://www.kdnuggets.com/2018/09/essential-math-data-science.html

http://stackoverflow.com/

https://stackoverflow.com/questions/31487264/numpy-eigenvalues-correct-but-eigenvectors-wro

Here is the book that we used for the class
https://www.springer.com/us/book/9783319747477