JYI Journal of Young Investigators

Science News

A Primer on Linear Algebra in Machine Learning

Sai Mannam

With the advent of many new machine learning packages and tools being open-sourced, there has been increased popularity in machine learning applications among various industries. From predicting the best routes for pharmaceutical drug delivery to gaining insights on stock movements using prior financial data, machine learning has been ever-expanding for the past few decades. Although there has been an increase in its utilization, there are still many who do not know the underlying math used in these algorithms.

WHY IS KNOWING THE MATH IMPORTANT?

With its vast applications, machine learning is affecting so many industries with such great magnitude. Many of these solutions entail different approaches and choosing the best one is up to the data scientist. The algorithm selection process should not simply be limited to its efficiency, but we must rather understand how and why the algorithm is doing what it is doing. For that reason, anyone with remote applications of machine learning in their industry should at least be able to interpret these algorithms based on the quality of math and parameter settings used.

WHAT DO I NEED TO KNOW ABOUT LINEAR ALGEBRA?

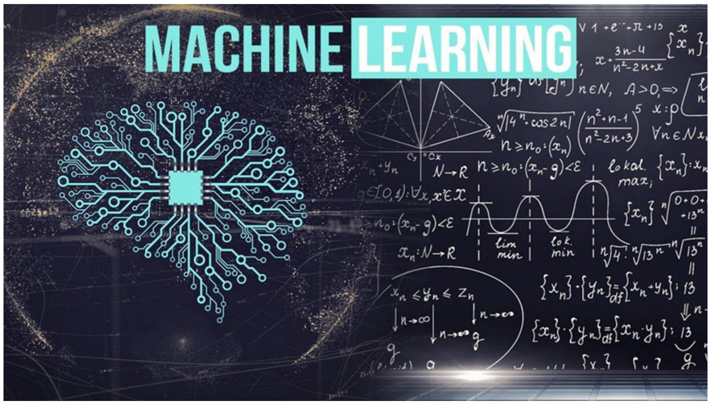
Although I can not condense a vast field such as Linear Algebra into a small section of an article, I can introduce how linear algebra is used in machine learning to pique the interest of readers.

Thus, we start the discussion with matrices, the central component to machine learning algorithms. A matrix is simply an array of numbers.

It can have as many columns and rows as we want. Usually, matrices are used as data representations. Many different mathematical manipulations can be performed on them, such as multiplication or division. They can also be transposed: switching the columns and rows with one another. We can even perform matrix algebra, where we attach matrices with variables and create linear models.

$$\mathbf{A} = \left[\begin{array}{ccccc} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn} \end{array}\right] \begin{array}{l} \\ \leftarrow \text{second row} \\ \\ \\ \\ \end{array}$$

$$\begin{array}{c} \uparrow \\ \text{third column} \end{array}$$



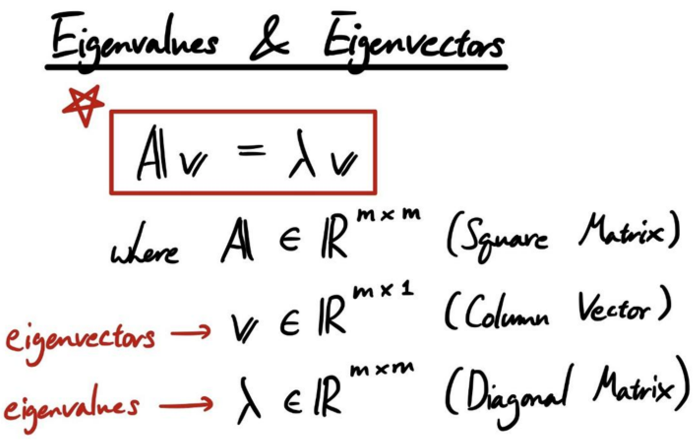
A matrix with one row or column is defined as a vector. These have special utilities for matrix analysis as well. We can take a set of vectors that represent different aspects of data and check for linear independence. Vectors are linearly independent only when we cannot write them as weighted combinations of previous vectors. This is important to check whether different parts of the data are related or independent of each other.

Another important concept in matrix algebra is eigenvectors and eigenvalues. This creates the foundation for principal component analysis, which is used to reduce dimensionality of datasets. Suppose we have matrix A with 3 columns and 3 rows. We take matrix A and multiply it with matrix B, which has 3 columns and 1 row. If this results in a vector multiplied by some constant (see image below), then we can consider the multiplied vector as an eigenvector of matrix A. The resulting constant is called the eigenvalue. This may seem quite convoluted, but at least having some idea of these concepts will help with understanding their broader uses in machine learning later on.

Although the simple definition of eigenvectors and eigenvalues do not really have a mathematical use for our purposes, their properties are impactful for their frequent applications in statistics.

THEN HOW DO WE APPLY LINEAR ALGEBRA?

The value in understanding the above concepts may be highlighted if we go through an example of their utility. My current research analyzes brain lesions and neurodegeneration with neuroimaging. Using machine learning and other statistical algorithms, I am able to analyze the data and more precisely



Eigenvalues & Eigenvectors

$\boxed{A v = \lambda v}$

where $A \in \mathbb{R}^{m\times m}$ (Square Matrix)

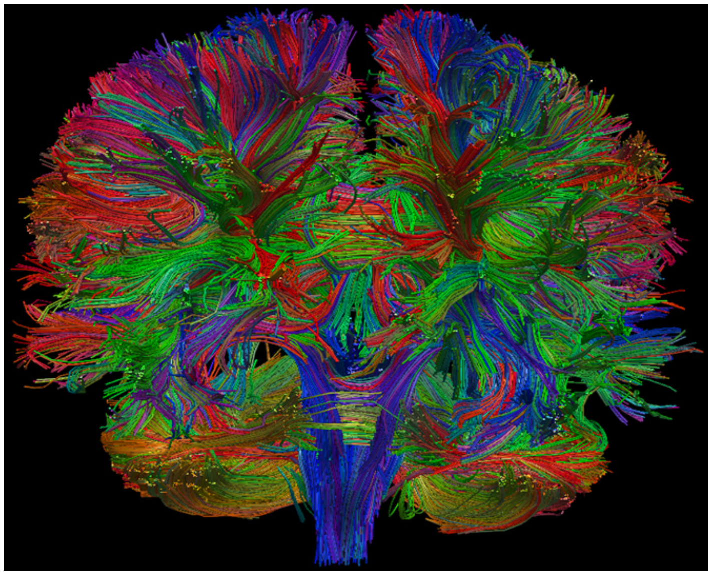
eigenvectors → $v \in \mathbb{R}^{m\times 1}$ (Column Vector)

eigenvalues → $\lambda \in \mathbb{R}^{m\times m}$ (Diagonal Matrix)

identify changes in the brain. Analyzing the neuroimaging data requires a shift in perspective to understand how the computer views the data.

The images from MRI, the modality that I use, are just bundles of pixels that we find meaning from. They provide information on the brain and various anatomical structures within it. In the simplest of terms, this information can be stored as discrete, smaller representations. The computer just identifies numbers of pixelated squares to represent an image. This image can be broken down into a matrix based on these specific numbers assigned from the light-dark spectrum. After we break this image into a pixel, if we add the time dimension as multiple images taken in succession, then we have 3-dimensional data. Each "pixel" in this dimension is called a voxel. Each voxel for the MRI can be stored as a matrix entity that we can manipulate as we discussed before.

The type of MRI data that I analyze is called diffusion tensor imaging (DTI), which measures the random motion of water molecules in a fluid to determine brain structure. The computer views image information as matrices of numbers organized in a certain manner. I can assess the direction of water molecules from something called b-vectors. While the images are being collected from a patient, the MRI machine can orient in different directions to collect information on the paths for water molecule diffusion. The b-vector then allows me to identify the volume of diffusion in the x-, y-, and z- directions (3-dimensional space). When I know the direction and level of diffusion of water molecules, then I can predict the gradient of diffusion along various signals.



This is where we can identify the application of eigenvalue and eigenvector properties. As we discussed, the flow direction of water molecules is represented by b-vectors. The direction of this energy is simply the eigenvector of the matrix we create. The magnitude of strength of the water molecule diffusion is the eigenvalue for our image matrix. Knowing this information, I can then manipulate and use their mathematical properties to analyze DTI with machine learning.

CONCLUDING REMARKS

Linear Algebra allows us to break down these complicated concepts into mathematical form. Having a universal language like this to analyze our data makes machine learning not only more accessible to other industries, but also more understandable. I hope that my explanation shared my passions as well as inspired some of the readers to want to learn more about the field and its relevant math. Machine learning is a beautiful way for us to make educated observations about how our world works. As more people incorporate it into more industries, we can share diverse insights with machine learning.

REFERENCES

https://www.youtube.com/watch?v=Rt6beTKDtqY

https://cnx.org/contents/fpkWedRh@2.3:-bqFVN8e@6/Linear-Algebra-Matrices

https://medium.com/sho-jp/linear-algebra-part-6-eigenvalues-and-eigenvectors-35365dc4365a

https://images.medicaldaily.com/sites/medicaldaily.com/files/2013/08/04/0/74/7436.png

https://royalsocietypublishing.org/doi/10.1098/rsta.2015.0202

https://pubmed.ncbi.nlm.nih.gov/28765809/

https://andysbrainbook.readthedocs.io/en/latest/MRtrix/MRtrix_Course/MRtrix_00_Diffusion_Overview.html