This is not going to be a full fledged course in linear algebra , we will cover the pieces which are relevant in context of machine learning . We will also develop intuition about how linear algebra , in context of ML starts with providing short hand representation of otherwise verbose descriptions of data and data operations . What we learn here , we will combine it with calculus in the next chapter to eventually learn optimisation in context of ML with data.

We will also look at some fundamental properties/jargon about matrices and matrix operations in the process; once we have established what do they represent in context of machine learning . Lets begin

Matrix Notation

Consider this layout of values

$$X = \begin{bmatrix} 25 & 70 & 175 \\ 30 & 65 & 168 \\ 35 & 80 & 180 \\ 40 & 75 & 170 \\ 28 & 58 & 160 \\ 45 & 85 & 178 \\ 22 & 55 & 162 \end{bmatrix}$$

We will call X a matrix with dimensions 7×3 , where 7 is number of rows and 3 is number of columns . At times we might also right X as $X^{7\times3}$ to represent it along with its dimensions . In general the notation would be $X^{rows\times columns}$. An individual element of matrix would be represented as X_{ij} , which essentially represents element in the i^{th} row and j^{th} column . for example , in the matrix given above $X_{23} = 168$

Transpose of a matrix is where , layout shown above is simply flipped in a way. Rows becomes columns and columns become rows . Transpose of X is represented as X^T .

$$X^T = \begin{bmatrix} 25 & 30 & 35 & 40 & 28 & 45 & 22 \\ 70 & 65 & 80 & 75 & 58 & 85 & 55 \\ 175 & 168 & 180 & 170 & 169 & 178 & 162 \end{bmatrix}$$

you can see here that

$$X_{ij} = X^T_{ji}$$

We will call a single column or single row of a matrix a vector [or a 1-D matrix, but vector would be a more common term]. For example

$$a = [\,25 \quad 70 \quad 175]$$

$a^{1\times3}$ or just a here is a row vector

https://github.com/lalitsachan/text-courses

$$b = \begin{bmatrix} 25 \\ 30 \\ 35 \\ 40 \\ 28 \\ 45 \\ 22 \end{bmatrix}$$

and $b^{7\times1}$ or just b here is column vector . Now lets see these in context of data as we will see during machine learning related operations. Do note that a row vector can be represented as transpose of a column vector . For example b^T here would be a row vector and a^T would be a column vector .

Vectors as row or columns of values

Consider this small data table

Name	Age	Weight (kg)	Height (cm)	100m Sprint Time (seconds)
John	25	70	175	11.2
Emma	30	65	168	12.1
Rajesh	35	80	180	13.5
Sarah	40	75	170	14.3
Priya	28	58	160	12.7
David	45	85	178	15.0
Ananya	22	55	162	11.8

https://github.com/lalitsachan/text-courses

Lets say we want to build a predictive equation for `Sprint Time` , with `Age` , `Weight` and `Height` as features . We can represent feature values for each row as a row vector [of dimension 1×3. For example , for `John` value of their feature values can be written as

$$\begin{bmatrix} 25 & 70 & 175 \end{bmatrix}$$

we can represent any given data column as a column vector , for example `Sprint Time` , the Target can be written as a column vector [of dimension 7×1]

$$\begin{bmatrix} 11.2 \\ 12.1 \\ 13.5 \\ 14.3 \\ 12.7 \\ 15.0 \\ 11.8 \end{bmatrix}$$

Dot Products of Two Vectors

Dot product of two vectors happens between a row and column vector and it is defined as follows

$$a = \begin{bmatrix} a_1 & a_2 & \cdots & a_t \end{bmatrix}$$

$$b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_t \end{bmatrix}$$

$$a \cdot b = a_1 * b_1 + a_2 * b_2 + \cdots a_t * b_t$$

Note couple of things here

- outcome of the dot product is a scalar value
- number of elements in both the vectors need to match

without being rigorous about it; lets say the predictive equation that we come up with for the data mentioned above is a linear model which goes like this :

$$\text{Sprint Time} = 5 + 0.5 * Age + 0.25 * Weight - 0.02 * Height$$

This for a particular person say `Priya` would be like this

$$\text{Sprint Time} = 5 + 0.5 * 28 + 0.25 * 58 - 0.02 * 160$$

we can write the same using dot product between two vectors where

$$X_i = \begin{bmatrix} 28 & 58 & 160 \end{bmatrix}$$

and

$$W = \begin{bmatrix} 0.50 \\ 0.25 \\ -0.02 \end{bmatrix}$$

and

$$\text{Sprint Time} = 5 + X_i \cdot W$$

Also note that outcome of dot product between row vector of dimension $1 \times k$ and column vector of dimension $k \times 1$ is a single scalar value.

Matrix Multiplication

You might remember the mechanics of a matrix multiplication from your school days [and if you dont, then we will discuss that anyway], goal of this section is to give you some rationale for why the matrix multiuplication is defined the way it is .

Matrix multiplication finds one of its origins in being a shorthand for propagating linear transformations.

Consider two variables x and y . Lets define g and h as linear combination of these as follows

https://github.com/lalitsachan/text-courses

$$g = a_1x + a_2y$$
$$h = b_1x + b_2y$$

lets write the same with matrices and vectors where we consider x ,y as one set of column vector from where we are arriving at g, h with linear transform driven by coefficients $\{a_1, a_2\}\ and\ \{b_1, b_2\}$. before that note that g is outcome of dot product between $\begin{bmatrix} a_1 & a_2 \end{bmatrix}$ and $\begin{bmatrix} x \\ y \end{bmatrix}$, and similarly h is outcome of dot product between $\begin{bmatrix} b_1 & b_2 \end{bmatrix}$ and $\begin{bmatrix} x \\ y \end{bmatrix}$. if we write those two operations together it looks like this.

$$\begin{bmatrix} g \\ h \end{bmatrix} = \begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Lets break down whats happening here , we are multiplying a 2×2 matrix $\begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix}$ with 2×1 column vector $\begin{bmatrix} x \\ y \end{bmatrix}$, which is resulting in a 2×1 vector $\begin{bmatrix} g \\ h \end{bmatrix}$. Consider the matrix $\begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix}$ as stack of row vectors. The outcome of multiplication between $\begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix}$ and $\begin{bmatrix} x \\ y \end{bmatrix}$ is result of dot product of each row vector in $\begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix}$, $\begin{bmatrix} a_1 & a_2 \end{bmatrix}$ and $\begin{bmatrix} b_1 & b_2 \end{bmatrix}$with the column vector $\begin{bmatrix} x \\ y \end{bmatrix}$. which gets us the result equivalent to $\begin{bmatrix} g \\ h \end{bmatrix} = \begin{bmatrix} a_1x + a_2y \\ b_1x + b_2y \end{bmatrix}$. Couple things to note

- number of elements in row vectors of the matrix $\begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix}$ need to be same as number of elements in the column vector $\begin{bmatrix} x \\ y \end{bmatrix}$
- It doesn't matter how many row vectors there are [we could have many more linear transformations such as g,h arising from x,y]
- we can generalise this observation to this : **we can only multiply a $n \times k$ matrix with a $k \times 1$ column vector** [k here represents number of elements in the row vectors and column vector; which needs to strictly match with each other]

Now consider that we define , another set of vectors m,n as subsequent linear transformations of g,h as follows

$$m = c_1g + c_2h$$
$$n = d_1g + d_2h$$

we can write this explicitly in terms of x,y , we can do that like this

$$m = c_1g + c_2h = (c_1a_1 + c_2b_1)x + (c_1a_2 + c_2b_2)y$$
$$n = d_1g + d_2h = (d_1a_1 + d_2b_1)x + (d_1a_2 + d_2b_2)y$$

instead of figuring this out in a tedious manner of starting with g,h in terms of x,y and then rearranging the terms to collect all the coefficients of x and y , separately; we can utilize our matrix multiplication notation used earlier

$$\begin{bmatrix} m \\ m \end{bmatrix} = \begin{bmatrix} c_1 & c_2 \\ d_1 & d_2 \end{bmatrix} \begin{bmatrix} g \\ h \end{bmatrix} = \begin{bmatrix} c_1 & c_2 \\ d_1 & d_2 \end{bmatrix} \begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} (c_1a_1 + c_2b_1) & (c_1a_2 + c_2b_2) \\ (d_1a_1 + d_2b_1) & (d_1a_2 + d_2b_2) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

https://github.com/lalitsachan/text-courses

Lets break down whats happening here . multiplication of $\begin{bmatrix} c_1 & c_2 \\ d_1 & d_2 \end{bmatrix}$ and $\begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix}$ generates $\begin{bmatrix} (c_1a_1 + c_2b_1) & (c_1a_2 + c_2b_2) \\ (d_1a_1 + d_2b_1) & (d_1a_2 + d_2b_2) \end{bmatrix}$. These are nothing but dot products of row vectors in $\begin{bmatrix} c_1 & c_2 \\ d_1 & d_2 \end{bmatrix}$, $\begin{bmatrix} c_1 & c_2 \end{bmatrix}$ and $\begin{bmatrix} d_1 & d_2 \end{bmatrix}$ with column vectors in $\begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix}$, $\begin{bmatrix} a_1 \\ b_1 \end{bmatrix}$ and $\begin{bmatrix} a_2 \\ b_2 \end{bmatrix}$. this implies couple of things

- When we are multiplying two matrices A ,B as AB ; number of elements in rows of A , needs to be same as number of elements in columns of B [as follows from dot product of two vectors discussed earlier]
- It doesn't matter how many rows are there in A [we could have had many more linear transformation such as m,n]
- it doesn't matter how many columns are there in B [We could have had many more base variables such as x ,y]
- But number of columns in A [which is same as number of elements in the row vectors in A] needs to strictly match with number of rows in B [which is same as number of elements in the column vectors in B]
- We can generalise this as : **We can multiply matrix of dimension $n \times k$ only with another matrix of dimension $k \times r$, the result is $n \times r$ matrix. The order of multiplication also matters, implying that $AB \neq BA$ and its very much possible that while AB is defined but BA might not be**

  Also observe from the calculations shown above that we can write individual elements of the matrix AB which is result of matrix multiplication of A and B as follows

  $$AB_{ij} = A_{i\cdot} \cdot B_{\cdot j}$$

  where $A_{i\cdot}$ is i^{th} row vector of A and $B_{\cdot j}$ is j^{th} column vector of B and $A_{i\cdot} \cdot B_{\cdot j}$ is dot product between the two .

  These notations are especially useful in defining neural network architectures in deep learning [that is not implying that they are not useful in traditional tabular ML]

https://github.com/lalitsachan/text-courses

Quick look at using this in context of an example data operation

Remember that we defined our so called model equation as :

$$\text{Sprint Time} = 5 + 0.5 * Age + 0.25 * Weight - 0.02 * Height$$

Now if we want to do this operation for all the observations we can write our entire data as stack of row vectors representing individual observations

$$X = \begin{bmatrix} 25 & 70 & 175 \\ 30 & 65 & 168 \\ 35 & 80 & 180 \\ 40 & 75 & 170 \\ 28 & 58 & 160 \\ 45 & 85 & 178 \\ 22 & 55 & 162 \end{bmatrix}$$

and we can multiply with the weight vector

$$W = \begin{bmatrix} 0.50 \\ 0.25 \\ -0.02 \end{bmatrix}$$

Multiplication of matrix X with column vector W , will give us outcome of that equation for all observation [after we add the bias term 5 to all], so we could write

$$\text{Predicted Sprint Times} = XW + \begin{bmatrix} 5 \\ \vdots \\ 5 \end{bmatrix}$$

https://github.com/lalitsachan/text-courses

A Square Matrix and Some Special Types

a square matrix is a matrix which has dimensions $n \times n$, essentially implying that number of rows are same as number of columns . In our ML data operations, you will find that square matrices show up all the time. Here is example of a 4×4 square matrix

$$\begin{bmatrix} 3 & 8 & 2 & 7 \\ 6 & 1 & 4 & 9 \\ 5 & 12 & 3 & 10 \\ 8 & 2 & 11 & 6 \end{bmatrix}$$

Diagonal Matrix

Square matrix with only non-zero elements being in its diagonal . Here is 4×4 , example

$$\begin{bmatrix} 3 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 6 \end{bmatrix}$$

Identity Matrix

Diagonal matrix with all the elements in the diagonal being 1. Here is 4×4 identity matrix

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Identity matrix acts in the same way as 1 acts in context of scalar multiplication . [As in you get the same scalar back when you multiply it with 1]. If you multiply a matrix with identity matrix [with appropriate dimension alignment], you will get the same matrix back . In general we can write

$$AI = A$$

where A is a square matrix with same dimensions as I , which is an identity matrix .

Symmetric Matrix

A symmetric matrix is square matrix which doesn't change on transposing . or in other words , a matrix A is said to be symmetric if $A^T = A$ and vice versa. here is an example

$$\begin{bmatrix} 4 & 7 & 3 & 5 \\ 7 & 9 & 6 & 4 \\ 3 & 6 & 2 & 8 \\ 5 & 4 & 8 & 7 \end{bmatrix}$$

Try transposing it and see what happens . Note that any kind of diagonal matrix will automatically be symmetric. **Another thing to note is a that we can consider a scalar to be a matrix with dimensions 1×1 which is by design a symmetric matrix. So for a scalar , $S^T = S$ is always true.**

https://github.com/lalitsachan/text-courses

Lower Triangle matrix

All elements above the diagonal are zeros. Here is an example

$$\begin{bmatrix} 5 & 0 & 0 & 0 \\ 3 & 6 & 0 & 0 \\ 7 & 2 & 4 & 0 \\ 9 & 1 & 8 & 3 \end{bmatrix}$$

Upper Triangle Matrix

All elements below the diagonal are zeroes. Here is an example

$$\begin{bmatrix} 5 & 3 & 7 & 9 \\ 0 & 6 & 2 & 1 \\ 0 & 0 & 4 & 8 \\ 0 & 0 & 0 & 3 \end{bmatrix}$$

Minor and Determinant of a Matrix

we will realize utility of these ideas in next section, for this section, lets just focus on definition and formulas for these terms .

For a 2×2 matrix:

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

The determinant is:

$$\det(A) = a_{11}a_{22} - a_{21}a_{12}$$

Minor: The minor of an element A_{ij} in a matrix A is the **determinant of the submatrix** formed by deleting the i^{th} row and j^{th} column from the original matrix. This submatrix is often denoted as M_{ij}.

A general formula of determinant for of any $n \times n$ matrix can be obtained with following formula using minors

The determinant can be obtained by expanding along any chosen row i or column j:

Expansion along row i:

$$\det(A) = \sum_{j=1}^{n} a_{ij}(-1)^{i+j}\det(M_{ij})$$

Expansion along column j:

$$\det(A) = \sum_{i=1}^{n} a_{ij}(-1)^{i+j}\det(M_{ij})$$

determinants of a matrix is invariant to row or column operations on the said matrix . For more details see this : [link]

Inverse of A Matrix

Division by a matrix is not something which makes much sense in usual context and is not really used. Instead of that we define something known as inverse for a square matrices [there exist something called pseudo-inverses for non-square matrices , but thats out of context for our discussion here] which is similar to the way inverse of a scaler is defined. Inverse of a scalar is defined as the number which when multiplied with that scalar, yields 1. Similar inverse of a square matrix is another matrix which when multiplied with it, yields an identity matrix.

for a 2×2 matrix

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

Inverse is defined as below

$$A^{-1} = \frac{1}{\det(A)} \begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix}$$

if you multiply A with A^{-1} you will get identity matrix I

$$AA^{-1} = I$$

Note that by design , inverse will not necessarily exist or be defined for every square matrix. If $det(A)$ is zero; A^{-1} does not exist . Inverse of higher dimension matrix can be determined with more complex heuristics which are not relevant to our discussion right here .

https://github.com/lalitsachan/text-courses

Eigen-Values and Eigen-Vectors of a Matrix

We have seen earlier that a matrix can be used to represent a linear transformation . In this section we will give it some geometric intuition and learn about eigen values in the process. These are very useful in terms of how linear algebra can help us model complex systems and processes. They also show up in some places in ML and thats why we are discussing them. Lets start by understanding some fundamentals first.

Basis

You might be familiar with the idea of writing a vector in 2D space [we will refer to it as $\mathbb{R}^2$] as

$$\begin{bmatrix}\alpha\\ \beta\end{bmatrix} = \alpha\hat{i} + \beta\hat{j}$$

where $\hat{i}$ represents the vector $\begin{bmatrix}1\\ 0\end{bmatrix}$, vector parallel to x-axis in the cartesian coordinate system and $\hat{j}$ representing vector $\begin{bmatrix}0\\ 1\end{bmatrix}$ parallel to y-axis. Lets understand the terminology which is used to represent the same idea with slightly different notations and names for the same.

Consider $\mathbb{R}^2$ as a vector space made of all possible $2D$ pair of points. Also considered as vectors originating from say $\{0, 0\}$. **A basis for a vector space is the smallest set of vectors which can generate that entire space and they are linearly independent** . For $\mathbb{R}^2$, that set of vectors can be $\{1, 0\}$ and $\{0, 1\}$. Take any other vector in $\mathbb{R}^2$, it can be written as linear combination of these two. So a generic vector in $\mathbb{R}^2$ can be defined as

$$\begin{bmatrix}\alpha\\ \beta\end{bmatrix} = \alpha\begin{bmatrix}1\\ 0\end{bmatrix} + \beta\begin{bmatrix}0\\ 1\end{bmatrix}$$

where α and β can be any set of real numbers. Also note that there is no possible real number say γ for which $\begin{bmatrix}1\\ 0\end{bmatrix} = \gamma\begin{bmatrix}0\\ 1\end{bmatrix}$ [meaning these two are linearly independent] . Basis of a vector space need not be unique . For example , another pair of vectors such as $\begin{bmatrix}2\\ 0\end{bmatrix}$ and $\begin{bmatrix}0\\ 3\end{bmatrix}$ can also be a basis vector , we will be able to write as above

$$\begin{bmatrix}\alpha\\ \beta\end{bmatrix} = \frac{\alpha}{2}\begin{bmatrix}2\\ 0\end{bmatrix} + \frac{\beta}{3}\begin{bmatrix}0\\ 3\end{bmatrix}$$

or consider $\begin{bmatrix}1\\ 1\end{bmatrix}$ and $\begin{bmatrix}1\\ -1\end{bmatrix}$ we can write

$$\begin{bmatrix}\alpha\\ \beta\end{bmatrix} = \frac{\alpha+\beta}{2}\begin{bmatrix}1\\ 1\end{bmatrix} + \frac{\alpha-\beta}{2}\begin{bmatrix}1\\ -1\end{bmatrix}$$

So only condition for set of vectors to be basis is that it needs to be smallest set from the vector space and they need to be linearly independent . Another misconception that people carry very often is that linear independence implies orthogonality. No, thats incorrect, linear independence simply implies that vectors shouldn't be aligned [same direction or 180° opposite], orthogonality by design implies linear independence but its not true vice versa.

https://github.com/lalitsachan/text-courses

so basis can be made up of vectors which are not orthogonal for example consider this linearly independent vectors as basis which are not orthogonal $\{\begin{bmatrix}1\\2\end{bmatrix}, \begin{bmatrix}2\\3\end{bmatrix}\}$. We can solve simple equations to write $\begin{bmatrix}\alpha\\\beta\end{bmatrix}$, in this basis.

$$\begin{bmatrix}\alpha\\\beta\end{bmatrix} = c_1\begin{bmatrix}1\\2\end{bmatrix} + c_2\begin{bmatrix}2\\3\end{bmatrix}$$
$$\alpha = c_1 + 2c_2$$
$$\beta = 2c_1 + 3c_2$$
$$c_1 = 2\beta - 3\alpha$$
$$c_2 = 2\alpha - \beta$$
$$\begin{bmatrix}\alpha\\\beta\end{bmatrix} = (2\beta - 3\alpha)\begin{bmatrix}1\\2\end{bmatrix} + (2\alpha - \beta)\begin{bmatrix}2\\3\end{bmatrix}$$

And once we have this basis , we will be able to write any arbitrary vector from that vector space as linear combination of the these basis vectors using the procedure shown above.

What happens to a vector in $\mathbb{R}^n$ when we multiply it with an $n \times n$ matrix

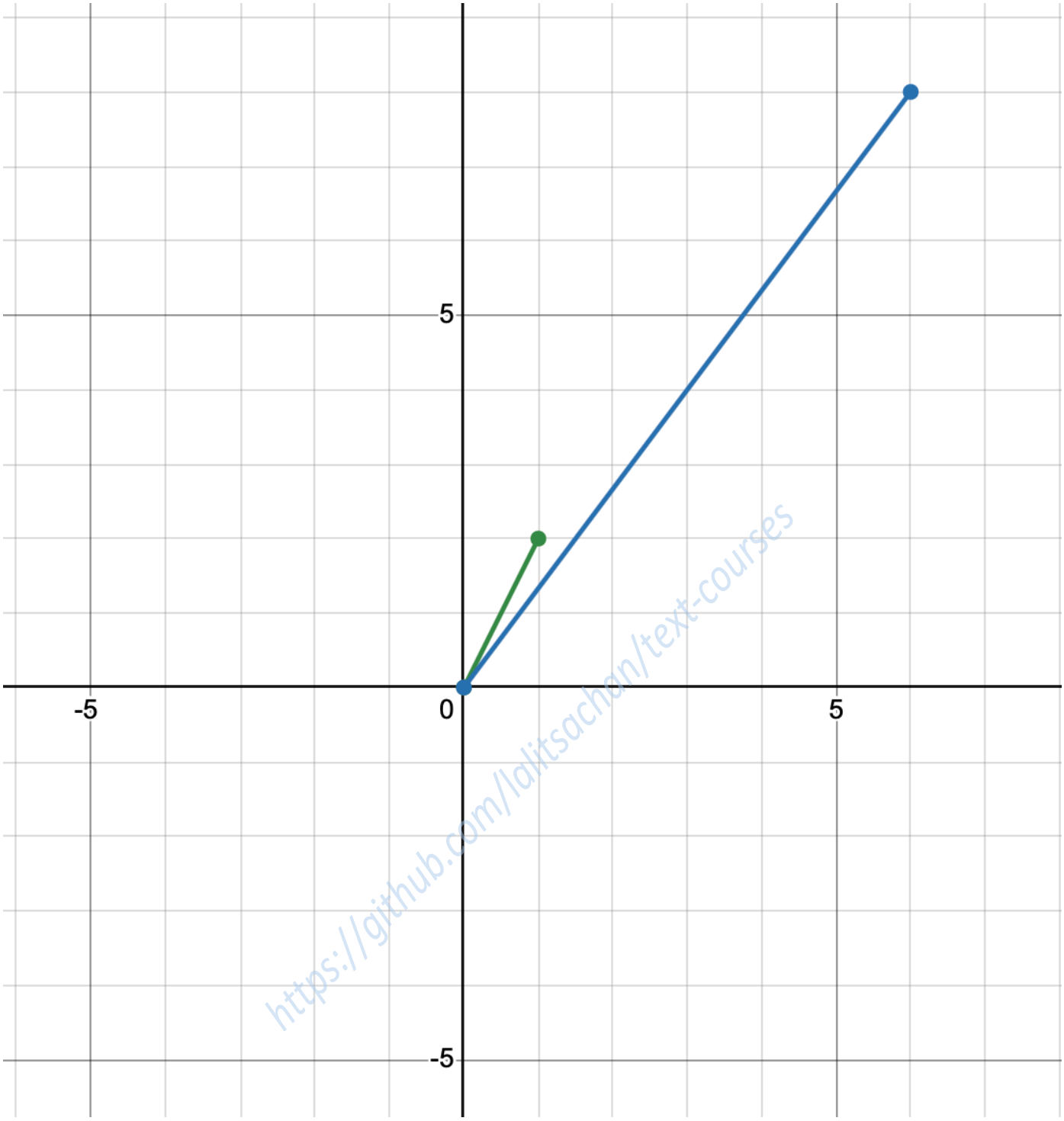
we will take the example of $\mathbb{R}^2$ here so that we can visualize whats going on, but the ideas can be extrapolated to higher dimension vector spaces without loss of generality.

Consider a vector $\begin{bmatrix}1\\2\end{bmatrix}$ being multiplied with a matrix $\begin{bmatrix}4 & 1\\2 & 3\end{bmatrix}$

$$\begin{bmatrix}4 & 1\\2 & 3\end{bmatrix} \times \begin{bmatrix}1\\2\end{bmatrix} = \begin{bmatrix}6\\8\end{bmatrix}$$

you can see that original vector $\begin{bmatrix}1\\2\end{bmatrix}$ shown with $green$ here , transforms into $\begin{bmatrix}6\\8\end{bmatrix}$ shown by $blue$. So a matrix multiplication essentially rotates and scales a vector .

https://github.com/lalitsachan/text-courses



https://github.com/lalitsachan/text-courses

but there is more to it, we know that the default basis that we use in $\mathbb{R}^2$ is $\{\begin{bmatrix}1\\0\end{bmatrix}, \begin{bmatrix}0\\1\end{bmatrix}\}$, simple reason being that its very easy to write any generic vector from $\mathbb{R}^2$ in this basis like this :

$$\begin{bmatrix}\alpha\\\beta\end{bmatrix} = \alpha\begin{bmatrix}1\\0\end{bmatrix} + \beta\begin{bmatrix}0\\1\end{bmatrix}$$

As mentioned earlier that vector spaces don't have unique basis, any vector in $\mathbb{R}^2$ for example can be written as a linear combination of any two other linearly independent vectors as basis. Here is where things become interesting , for matrix $\begin{bmatrix}4 & 1\\2 & 3\end{bmatrix}$ there exist two special vectors $\{\begin{bmatrix}1\\1\end{bmatrix}, \begin{bmatrix}1\\-2\end{bmatrix}\}$, such that if we re-wrote a vector in this basis, when multiplied by the matrix $\begin{bmatrix}4 & 1\\2 & 3\end{bmatrix}$, it will only scale by a factor of 5 in the

direction represented by $\begin{bmatrix}1\\1\end{bmatrix}$ and by a factor of 2 in the direction represented by $\begin{bmatrix}1\\-2\end{bmatrix}$. Lets understand that by an example . lets say we want to re-write a generic vector $\begin{bmatrix}\alpha\\\beta\end{bmatrix}$ in the basis $\{\begin{bmatrix}1\\1\end{bmatrix}, \begin{bmatrix}1\\-2\end{bmatrix}\}$

$$\begin{bmatrix}\alpha\\\beta\end{bmatrix} = c_1\begin{bmatrix}1\\1\end{bmatrix} + c_2\begin{bmatrix}1\\-2\end{bmatrix}$$
$$\alpha = c_1 + c_2$$
$$2\beta = c_1 - 2c_2$$
$$c_1 = \frac{2\alpha+\beta}{3}$$
$$c_2 = \frac{\alpha-\beta}{3}$$

using the formula above we can write $\begin{bmatrix}1\\2\end{bmatrix}$ as $\frac{4}{3}\begin{bmatrix}1\\1\end{bmatrix} - \frac{1}{3}\begin{bmatrix}1\\-2\end{bmatrix}$ and when we multiply this by $\begin{bmatrix}4&1\\2&3\end{bmatrix}$, this will simply become $5 \times \frac{4}{3}\begin{bmatrix}1\\1\end{bmatrix} - 2 \times \frac{1}{3}\begin{bmatrix}1\\-2\end{bmatrix} = \begin{bmatrix}6\\8\end{bmatrix}$

you can try this with any other random vector from $\mathbb{R}^2$. Lets take another example , consider vector $\begin{bmatrix}2\\3\end{bmatrix}$, using the formula derived above this can be written in the basis $\{\begin{bmatrix}1\\1\end{bmatrix}, \begin{bmatrix}1\\-2\end{bmatrix}\}$ as follows :

$$\begin{bmatrix}2\\3\end{bmatrix} = \frac{7}{3}\begin{bmatrix}1\\1\end{bmatrix} - \frac{1}{3}\begin{bmatrix}1\\-2\end{bmatrix}$$

so if we multiply $\begin{bmatrix}2\\3\end{bmatrix}$ with matrix $\begin{bmatrix}4&1\\2&3\end{bmatrix}$, it will simply scale by 5 and 2 along the directions given by the basis $\{\begin{bmatrix}1\\1\end{bmatrix}, \begin{bmatrix}1\\-2\end{bmatrix}\}$.

$$\begin{bmatrix}4&1\\2&3\end{bmatrix} \times \begin{bmatrix}2\\3\end{bmatrix} = 5 \times \frac{7}{3}\begin{bmatrix}1\\1\end{bmatrix} - 2 \times \frac{1}{3}\begin{bmatrix}1\\-2\end{bmatrix}$$
$$= \frac{35}{3}\begin{bmatrix}1\\1\end{bmatrix} - \frac{2}{3}\begin{bmatrix}1\\-2\end{bmatrix}$$
$$= \begin{bmatrix}11\\13\end{bmatrix}$$

That begs the question, how did we find these vectors for the matrix $\begin{bmatrix}4&1\\2&3\end{bmatrix}$. consider that v_1 and v_2 are the vectors which form such a basis for the multiplication by a matrix A. It implies that if you multiplied A with either v_1 or v_2 it will just scale them and not change their directions because the result needs to be written again in terms of v_1 and v_2 and they are linearly independent . which tells us that for such vectors following will be true.

$$Av = \lambda v$$
$$(A - \lambda I)v = 0$$

https://github.com/lalitsachan/text-courses

where λ is some real number . lets solve this for the matrix $\begin{bmatrix}4 & 1\\2 & 3\end{bmatrix}$

$$\left(\begin{bmatrix}4 & 1\\2 & 3\end{bmatrix}-\lambda\begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}\right)v=0$$
$$\left(\begin{bmatrix}4-\lambda & 1\\2 & 3-\lambda\end{bmatrix}\right)v=0$$
$$det\left(\begin{bmatrix}4-\lambda & 1\\2 & 3-\lambda\end{bmatrix}\right)=(4-\lambda)(3-\lambda)-2$$
$$\lambda^2-7\lambda+10=0$$
$$(\lambda-5)(\lambda-2)=0$$
$$\lambda\in\{5,2\}$$

let v be of the form $\begin{bmatrix}a\\b\end{bmatrix}$

$$\text{for } \lambda{=}5$$
$$\begin{bmatrix}-1 & 1\\2 & -2\end{bmatrix}\begin{bmatrix}a\\b\end{bmatrix}=0$$
$$-a+b=0\cdots(1)$$
$$2a-2b=0\cdots(2)$$
$$\text{solving } (1) \text{ and } (2) \text{ we get } a=b$$

which makes $\begin{bmatrix}1\\1\end{bmatrix}$ one such vector with corresponding $\lambda=5$. similarly for $\lambda=2$ we obtain vector $\begin{bmatrix}1\\-2\end{bmatrix}$

Formally these are called eigen vectors and corresponding eigen values of a matrix. In our example $\{\begin{bmatrix}1\\1\end{bmatrix},\begin{bmatrix}1\\-2\end{bmatrix}\}$ are eigenvectors of matrix $\begin{bmatrix}4 & 1\\2 & 3\end{bmatrix}$ with corresponding eigen values $\{5,2\}$.

Note that the characteristic equation that we solved for λ need not always have real solution, in that case the matrix in question does not have eigenvectors form a basis along which the matrix only scales. Complex number solutions for λ do have meaning in many places in physics but not so relevant for us , so we will not expand on that much here.

Eigen values of a real valued symmetric matrix

for this discussion i need to remind you about conjugate of complex numbers. Say there is a complex number $z=a+bi$ with a and b being some non-zero real numbers . Then their conjugate is written as $\bar{z}=a-bi$. Now if $z=\bar{z}$, it implies that $b=0$, which can be further understood to be indication of that z is not complex . Also if we have $z_1=z_2$, then $\overline{z_1}=\overline{z_2}$

Now lets examine if its possible for a real valued symmetric matrix to have complex eigen values and in turn complex eigenvectors. Lets assume that for a real valued matrix A such that $A^T=A$ we have a complex eigenvalue λ and a corresponding complex eigenvector v . Since these are eigenvectors and eigenvalues of A we can write

https://github.com/lalitsachan/text-courses

$$
\begin{aligned}
Av &= \lambda v \\
&\text{since } A \text{ is real valued } A = \overline{A} \\
\overline{Av} &= \overline{\lambda v} \\
A\bar{v} &= \bar{\lambda}\bar{v} \cdots (1) \\
&\text{lets multiply both sides with } \bar{v}^T \\
\bar{v}^T Av &= \bar{v}^T \lambda v \\
&\text{since } A \text{ is symmetric we have } A^T = A \\
\bar{v}^T A^T v &= \bar{v}^T \lambda v \\
&\text{since } (AB)^T = B^T A^T \text{, we can write} \\
(A\bar{v})^T v &= \bar{v}^T \lambda v \\
&\text{using } (1) \text{ here we get} \\
(\bar{\lambda}\bar{v})^T v &= \lambda \bar{v}^T v \\
\bar{\lambda}\bar{v}^T v &= \lambda \bar{v}^T v \\
&\text{since } \bar{v}^T v \neq 0 \text{ ,it leads to:} \\
\bar{\lambda} &= \lambda
\end{aligned}
$$

which we have seen earlier implies that λ is a real number after all. so the outcome means that **real valued symmetric matrices always have real eigenvalues and vectors**

Another fact is that for distinct eigenvalues say λ_1 and λ_2 such that $\lambda_1 \neq \lambda_2$, the corresponding eigenvectors v_1 and v_2 are orthogonal [$v_2^T v_1 = 0$]. lets see how that is true.

$$
\begin{aligned}
Av_1^T &= \lambda_1 v_1 \\
Av_2^T &= \lambda_2 v_2 \\
v_2^T Av_1 &= v_2^T \lambda_1 v_1 \\
v_2^T A^T v_1 &= v_2^T \lambda_1 v_1 \\
(Av_2)^T v_1 &= v_2^T \lambda_1 v_1 \\
\lambda_2 v_2^T v_1 &= \lambda_1 v_2^T v_1 \\
(\lambda_1 - \lambda_2) v_2^T v_1 &= 0 \\
&\text{since we have established that } \lambda_1 \neq \lambda_2 \\
v_2^T v_1 &= 0
\end{aligned}
$$

https://github.com/lalitsachan/text-courses

so a complete statement would be that **real valued symmetric matrices have real eigenvalues and orthogonal eigenvectors for distinct eigenvalues**

while we will encounter eigen values/vectors in ML discussion, however this discussion is incomplete without some excellent use cases discussed here:

- https://setosa.io/ev/eigenvectors-and-eigenvalues/
- https://math.stackexchange.com/questions/23312/what-is-the-importance-of-eigenvalues-eigenvectors

This much discussion should be enough for us to sail through or develop other mathematical ideas in context as far as ML is concerned. Lets get to the calculus discussion now.

https://github.com/lalitsachan/text-courses