MA5832: Data Mining & Machine Learning

Collaborate Week 1: Intro & Linear Algebra Review

Martha Cooper, PhD

JCU Masters of Data Science

2021-08-31

Housekeeping

- Collaborates = Tuesdays 6-7:30pm

For my Collaborate Sessions, you can get the slides & R code for each week on Github:

https://github.com/MarthaCooper/MA8532

Today's Goals

- Overview of topics covered in MA5832: Study Plan, Assessments & Expectations
- Vectors, Matrices & Linear Algebra
  - Matrix addition & multiplication
  - Computing the determinant for 2x2 and 3x3 matrices
  - Eigenvalues & Eigenvectors

MA5832: Data Mining & Machine Learning

MA5832 Study Plan

```
##   Weeks                 Collaborate_Topics
## 1     1 MA5832 Overview & Linear Algebra
## 2     2       Probability & Optimisation
## 3     3            Tree based regression
## 4     4         Support Vector Machines
## 5     5                   Neural Network
## 6     6                     Capstone Q&A
```

Assessments

Time management is important!

Assessments 1 due 12/09/21 (25%) - *Week 2 topics*

Assessments 2 due 26/09/21 (35%) - *Week 3 & 4 topics*

Assessments 3 (Capstone) due 13/10/21 (40%) *Bring everything together*

Expectations

1. Independent study is required.
2. Extensions
    - Read Section 4 of Course Outline
    - Requests must be emailed to Dr. David Donald before the deadline (unless it is an emergency)
3. Assessments 1 & 2 Submission Details
    - Must be submitted in PDF format
    - Can be written in .Rmd or word processor
    - Appendix with R code must be attached at the end of the same PDF document.
4. Questions?
    - MA5832 Discussion board: Nayar Sultana, Corey Lammie, Chinedu Ossai and myself
5. Questions about collaborates?
    - Email: martha.cooper@jcu.edu.au
    - MA5832 Discussion board: Tuesday & Wednesday

Vectors, Matrices & Linear Algebra

- Understand some basic concepts of linear algebra (revision…!)

Vectors

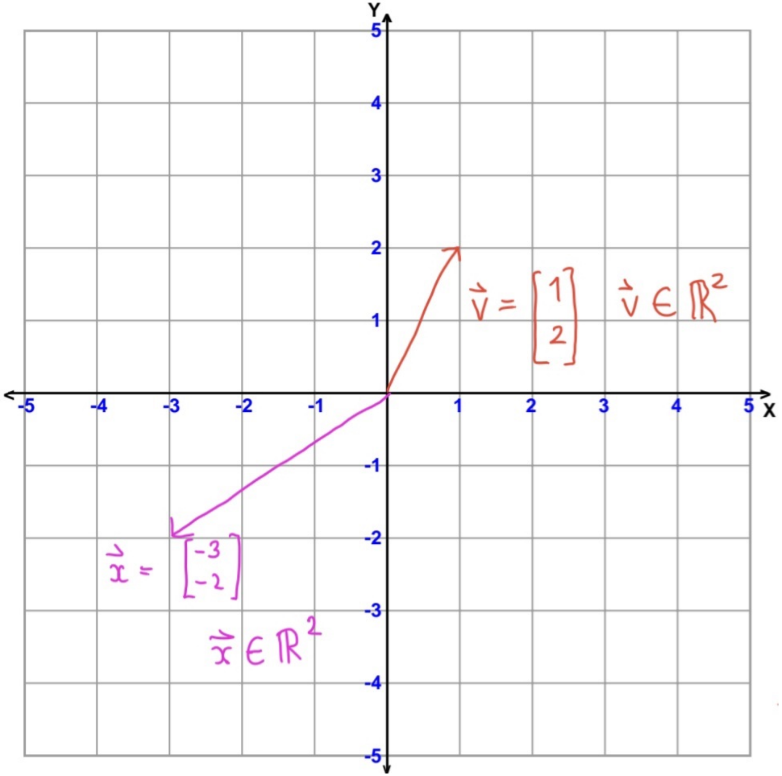
A vector is an array of numbers written as a column and enclosed by square brackets

$$\boldsymbol{v} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix}$$

This vector, $\boldsymbol{v}$, contains $\boldsymbol{m}$ elements. If each element of $\boldsymbol{v}$ is in $\mathbb{R}$, $\boldsymbol{v} \in \mathbb{R}^m$

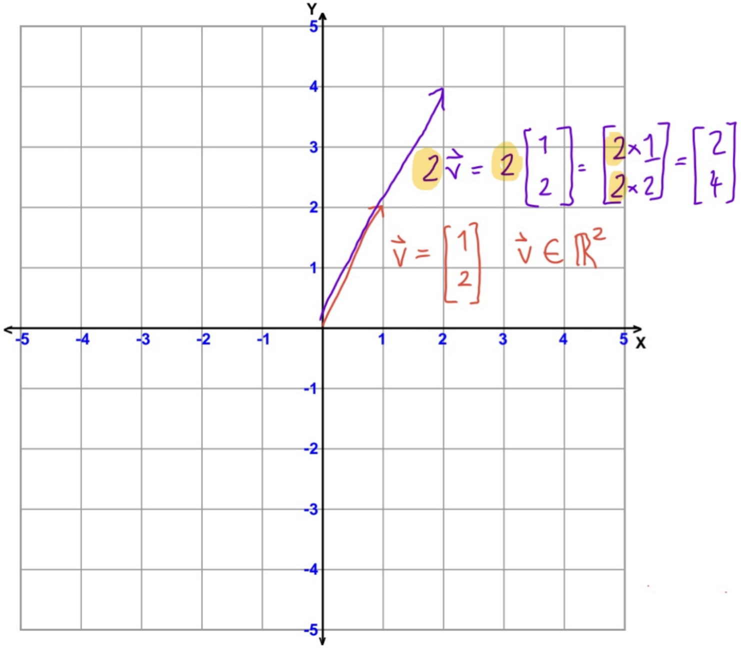
Vectors

Vectors can be represented geometrically



Scalars

A scalar is a number *e.g.* x



Vectors & Scalars in R

$$\boldsymbol{v} = \begin{bmatrix} 3 \\ 5 \\ 2 \end{bmatrix}$$

```r
v <- c(3,5,2) #defining a vector
v
```

```
## [1] 3 5 2
```

$$\boldsymbol{g} = 3\boldsymbol{v}$$

```r
g <- 3*v #multiplying a vector by a scalar
g
```

```
## [1]  9 15  6
```

Matrices

A matrix is a rectangular array of numbers, arranged in rows and columns. An $\boldsymbol{n} \times \boldsymbol{p}$ matrix has $\boldsymbol{n}$ rows and $\boldsymbol{p}$ columns

$$\boldsymbol{X} = \begin{bmatrix} x_{1,1} & x_{2,1} & \cdots & x_{1,p} \\ x_{2,1} & x_{2,1} & \cdots & x_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n,1} & x_{n,2} & \cdots & x_{n,p} \end{bmatrix}$$

If $\boldsymbol{x}_{i,j} \in \mathbb{R}$, then $\boldsymbol{X} \in \mathbb{R}^{n \times p}$

- Rows = Samples/Observations
- Columns = Variables/Factors/Predictors

Setting up matrices in R

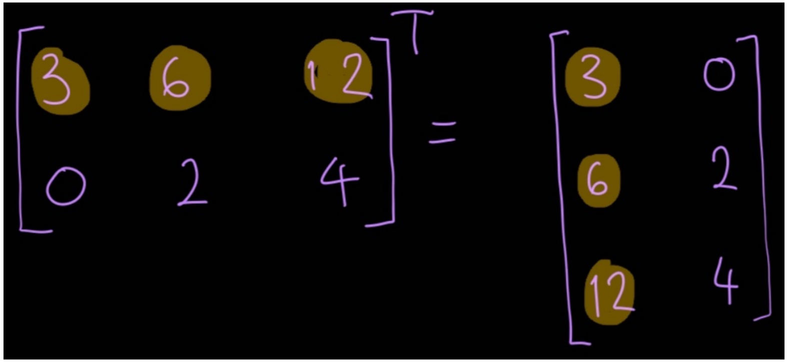
```
m <- matrix(c(1:9), nrow = 3, ncol = 3, byrow = T)
m
```

```
##      [,1] [,2] [,3]
## [1,]    1    2    3
## [2,]    4    5    6
## [3,]    7    8    9
```

Matrix Concepts

Transpose

$$A^T$$



Matrix Concepts

Inverse

$$\boldsymbol{A}^{-1}$$

With numbers:

$$\frac{5}{1} \times \frac{1}{5} = 1$$

We can do the same thing with matrices:

$$\boldsymbol{A}\boldsymbol{A}^{-1} = \boldsymbol{I}$$

Where $\boldsymbol{I}$ is the Identity matrix - the 1 equivalent of a matrix

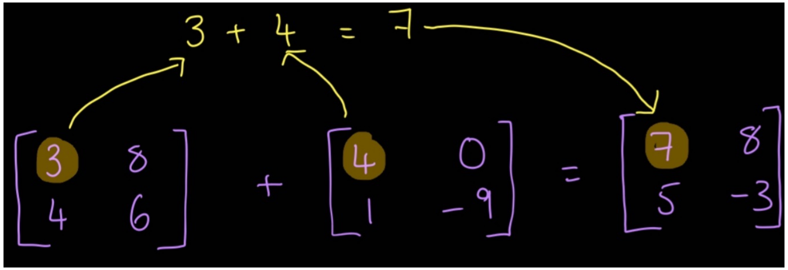
$$I_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Matrix Addition

$$\boldsymbol{A} + \boldsymbol{B}$$

Add the numbers in the matching positions. (& subtraction is the same, because it is the addition of a negative matrix

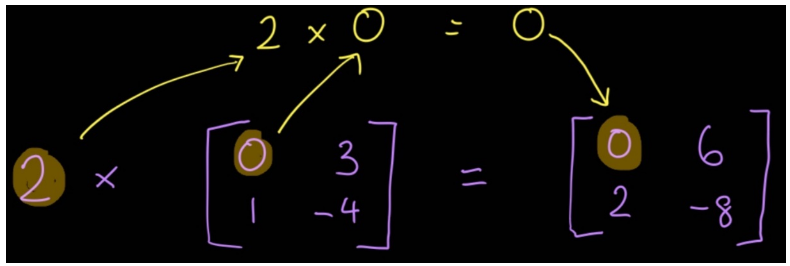
$$\boldsymbol{A} + (-\boldsymbol{B})$$



Note: The matrices must be the same size

Multiplying a matrix by a scalar

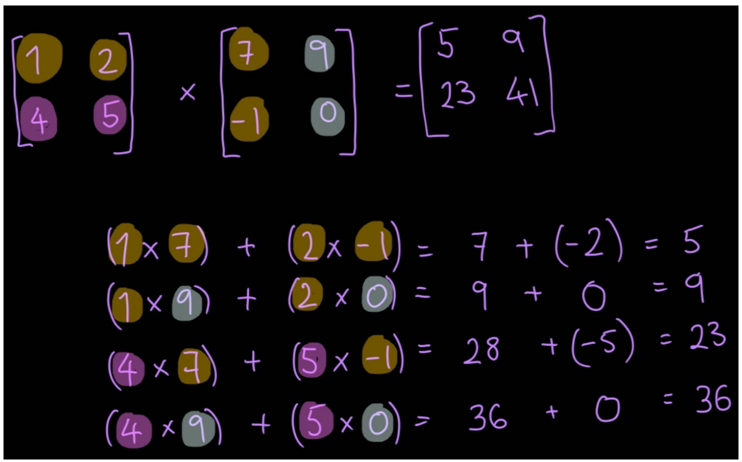
$$xA$$

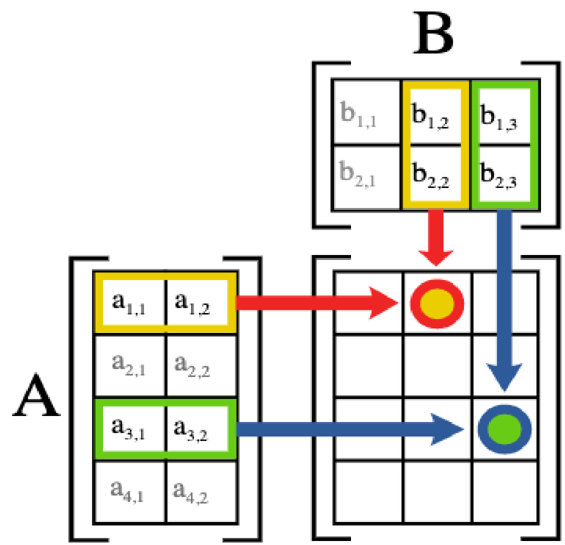


Matrix Multiplication

$$\boldsymbol{AB}$$

Take the dot product

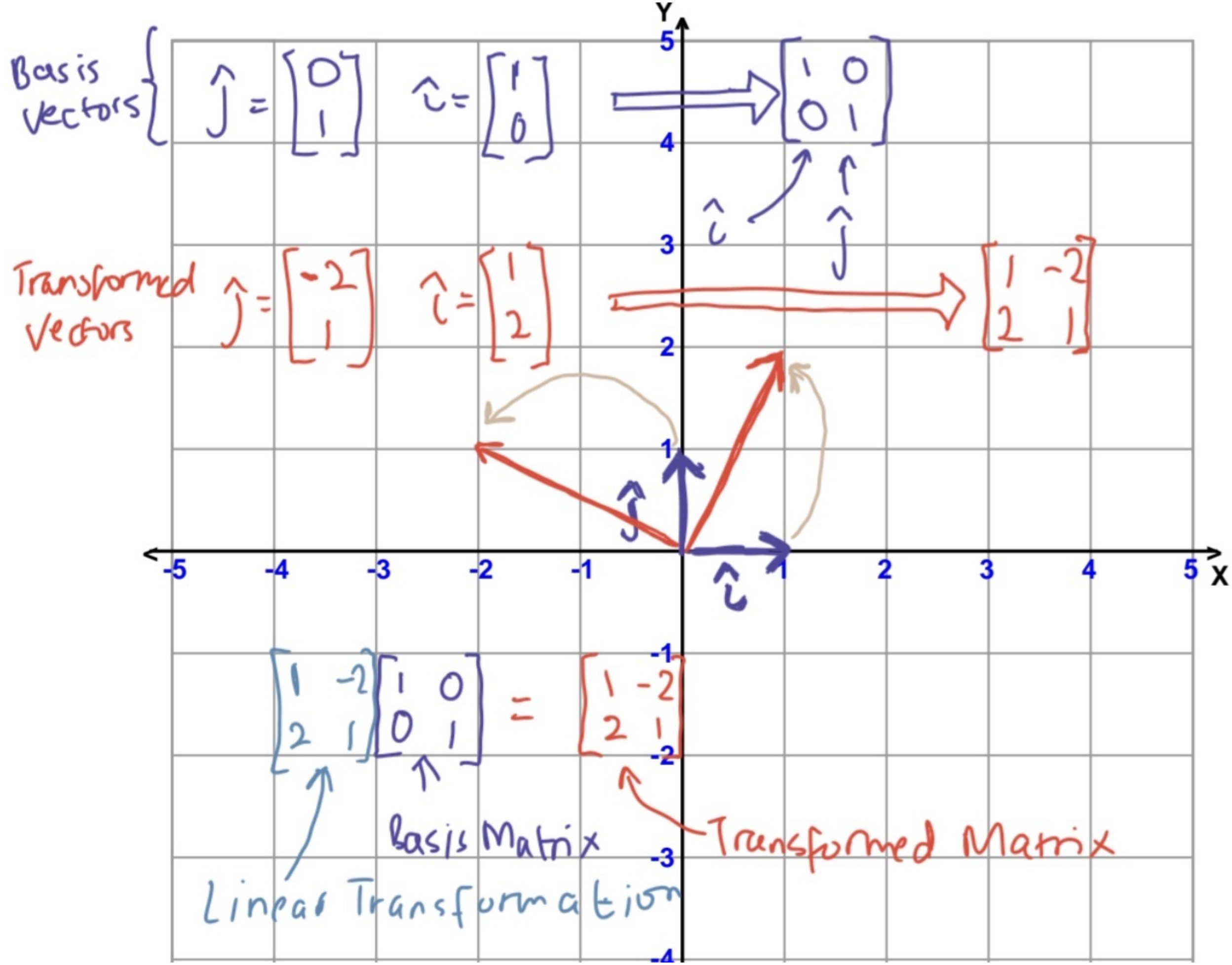




The number of columns in the left matrix must equal the number of row in the right matrix

Visualising Matrix Multiplication

Matrix multiplication is a linear transformation which we can see geometrically



Summary of Addition and Multiplication

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{bmatrix} ; \quad B = \begin{bmatrix} b_{1,1} & b_{1,2} \\ b_{2,1} & b_{2,2} \end{bmatrix}$$

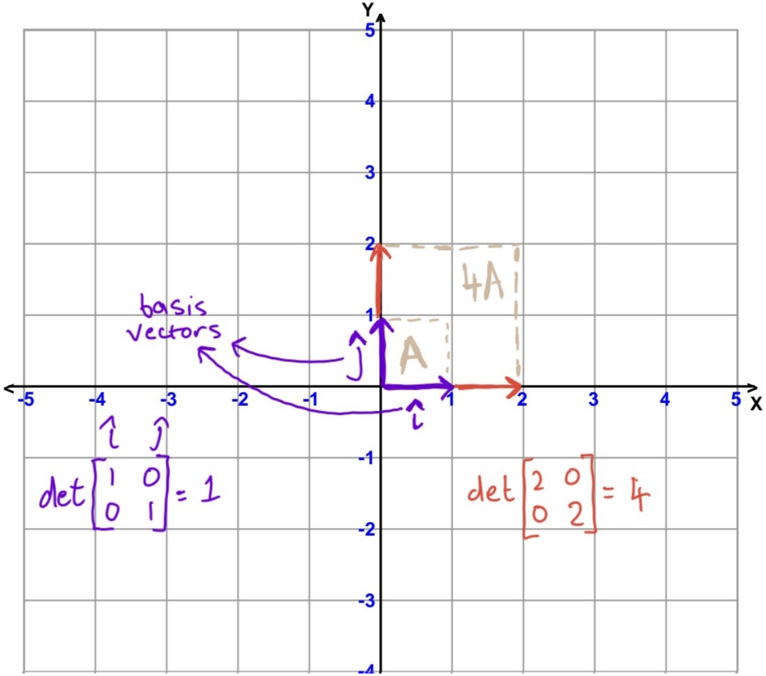
$$A + B = \begin{bmatrix} a_{1,1} + b_{1,1} & a_{1,2} + b_{1,2} \\ a_{2,1} + b_{2,1} & a_{2,2} + b_{2,2} \end{bmatrix}$$

$$A \times B = \begin{bmatrix} a_{1,1} \times b_{1,1} + a_{1,2} \times b_{2,1} & a_{1,1} \times b_{1,2} + a_{1,2} \times b_{2,2} \\ a_{2,1} \times b_{1,1} + a_{2,2} \times b_{2,1} & a_{2,1} \times b_{1,2} + a_{2,2} \times b_{2,2} \end{bmatrix}$$

$$A \odot B = \begin{bmatrix} a_{1,1} \times b_{1,1} & a_{1,2} \times b_{1,2} \\ a_{2,1} \times b_{2,1} & a_{2,2} \times b_{2,2} \end{bmatrix}$$

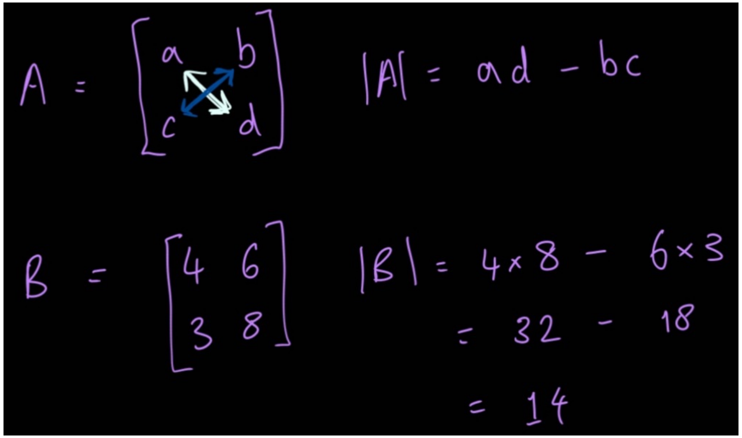
Determinant of a Matrix

Determinant - describing how linear transformations change area or volume. Also useful for solving linear equations and changing variables integrals.

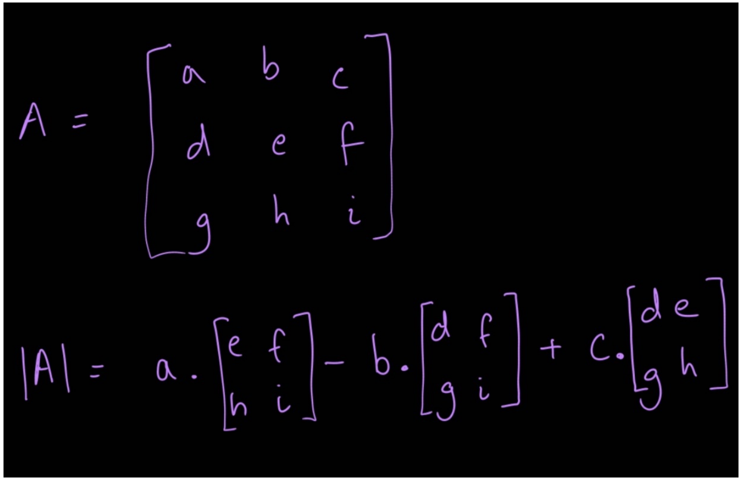


Computing the Determinant

2x2 matrices



3x3 matrices



Matrix Concepts in R

```r
# a and b are two square matrices

a + b # Addition
a %*% b # Multiplication
t(a) # Transpose
det(a) # Determinant
solve(a) # Inverse
```

Test these for yourself by hand and then using R

```r
h <- matrix(c(2,7,19,4), nrow = 2, byrow = F)
g <- matrix(c(1,3,4,2), nrow = 2, byrow = F)
```

Solving equations using matrices

Matrices make solving linear equations easier/faster

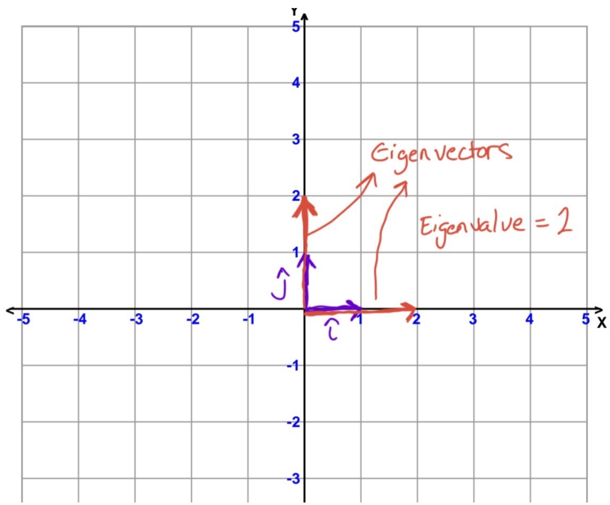
$$
\begin{aligned}
x + y + z &= 6 \\
2y + 5z &= -4 \\
2x + 5y - z &= 27
\end{aligned}
$$

$$
\begin{bmatrix} 1 & 1 & 1 \\ 0 & 2 & 5 \\ 2 & 5 & -1 \end{bmatrix}
\begin{bmatrix} x \\ y \\ z \end{bmatrix}
=
\begin{bmatrix} 6 \\ -4 \\ 27 \end{bmatrix} \quad \ldots
$$

Eigenvalues and Eigenvectors

In (some) linear transformations, there are vectors that don't change direction and are only scaled (stretched or shrunk) within their own span.

Eigenvectors are the vectors that remain pointing in the same direction.
Eigenvalues are the scalars that the eigenvectors are stretch/shrunk by.



Eigenvectors and Eigenvalues

Almost all vectors change direction when multiplying a matrix, $\boldsymbol{A}$. Eigenvectors, $\boldsymbol{x}$, are certain vectors that have the same direction as $\boldsymbol{A}x$. The Eigenvalue, $\boldsymbol{\lambda}$ is the scalar by which $\boldsymbol{x}$ is *stretched*, *shrunk*, *reversed* or *remained unchanged* when multiplied by $\boldsymbol{A}$.

$$\boldsymbol{A}\boldsymbol{x} = \lambda \boldsymbol{x}$$

We can find eigenvectors and eigenvalues of $\boldsymbol{A}$ by setting the determinant of $\boldsymbol{A} - \lambda \boldsymbol{I}$ to be 0.

$$det \mid (\boldsymbol{A} - \lambda \boldsymbol{I}) \mid = 0$$

Eigenvector and Eigenvalues in R

```
mat <- matrix(c(0.5, 0.5, 0.5, 0.5), byrow = T, nrow = 2)
mat
```

```
##      [,1] [,2]
## [1,]  0.5  0.5
## [2,]  0.5  0.5
```

```
eigen(mat)
```

```
## eigen() decomposition
## $values
## [1] 1 0
##
## $vectors
##           [,1]       [,2]
## [1,] 0.7071068 -0.7071068
## [2,] 0.7071068  0.7071068
```

Take what you have learned today and be able to:

- Perform linear regression using matrices (linear regression from scratch without using `lm`)
- Calculate eigenvalues and eigenvectors (by hand)

Extra reading

Very stuck?

- Essence of Linear Algebra by 3Blue1Brown
- Maths is fun Intro to Matrices
- Lumen Learning Boundless Algebra Intro to Matrices
- Maths is Fun Eigenvectors and Eigenvalues

Further Reading

- Chapter 2 Deep Learning by Goodfellow, Bengio & Courville

References

- Chapter 2 Deep Learning by Goodfellow, Bengio & Courville
- Maths is fun Intro to Matrices
- Lumen Learning Boundless Algebra Intro to Matrices
- Dr Kelly Trinh MA5832 Week 1:Linear Alegbra

Slides

- xaringhan, xaringanthemer, remark.js, knitr, R Markdown