

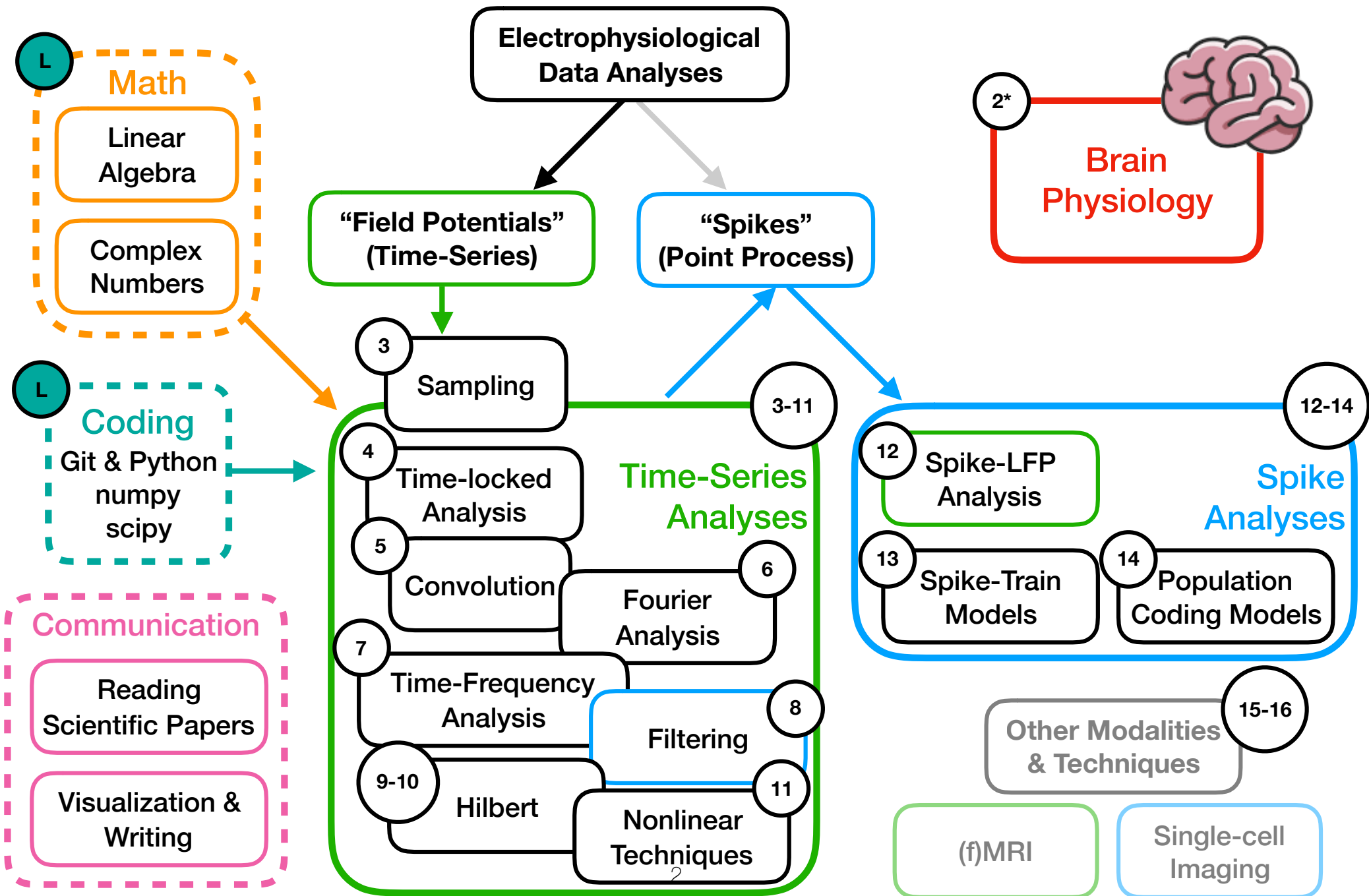
Math & Python Bootcamp

Lab 1

July 1, 2019



Course Outline: Road Map



1. Get (re)acquainted with GitHub & python
2. Vectors and dot product
3. Complex numbers



Fork & Clone Class Repo

<https://github.com/rdgao/COGS118C>



rdgao / COGS118C

Watch 0

Star 0

Fork 0

Code

Issues 0

Pull requests 0

Projects 0

Wiki

Security

Insights

Settings

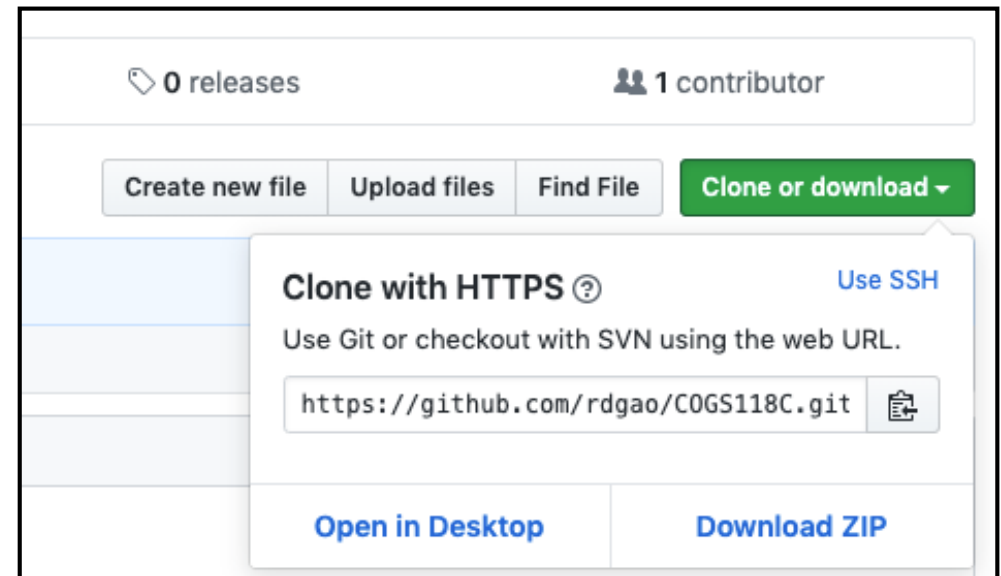
COGS118C [Neural Signal Processing] @ UCSanDiego

Edit

[Manage topics](#)

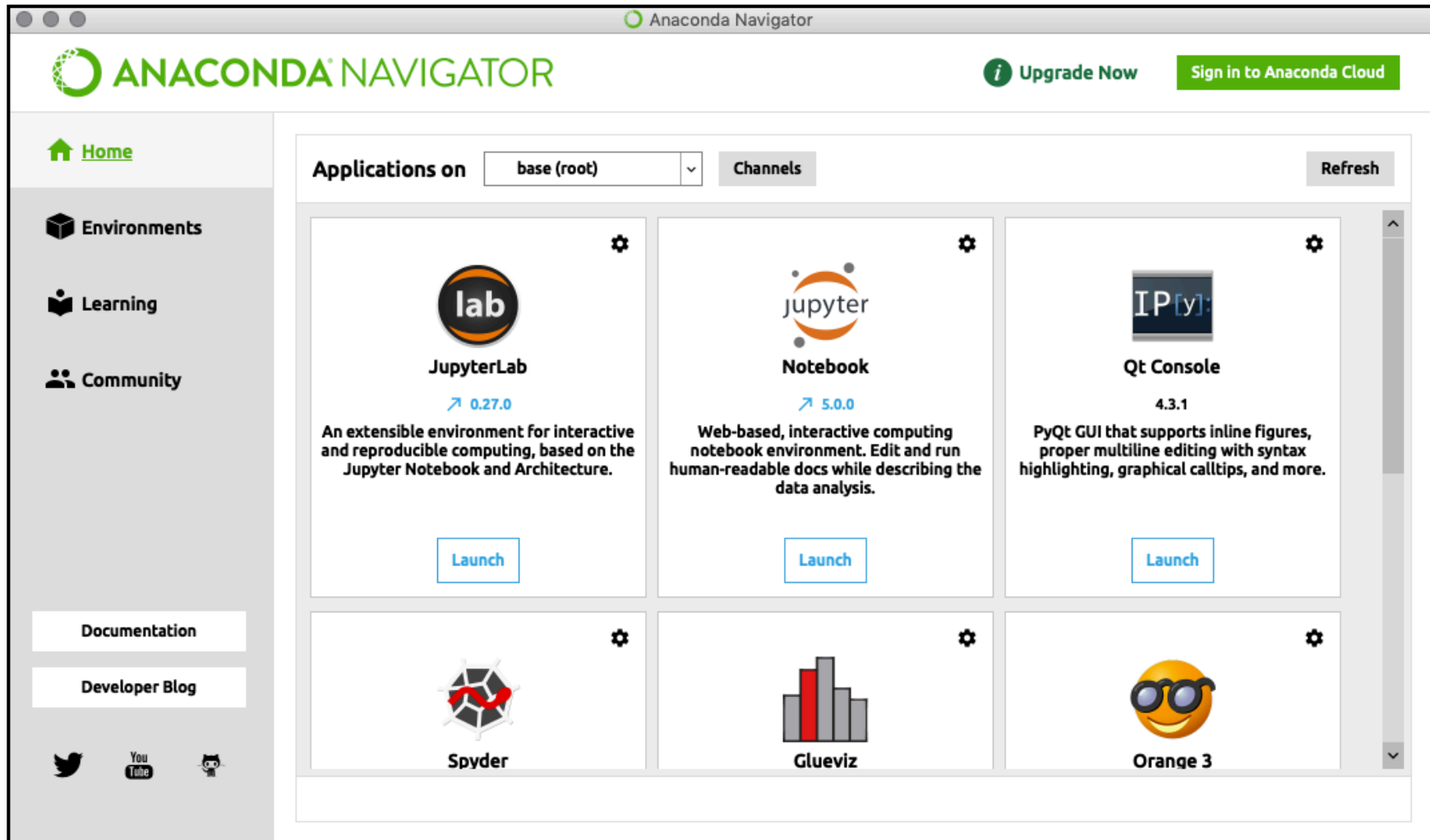
Now go to your GitHub account, where the forked repo should be, and clone a local version (on your laptop).

You can use git or GitHub Desktop




Open Jupyter Notebook

From **Anaconda Navigator**, or directly from Terminal (> jupyter notebook)



Some Basic Python Stuff



COGS 18 - Introduction To Python

Home

Materials

00-Introduction

← TOGGLE SIDEBAR



Download Notebook

Introduction to Python

Shannon Ellis

Important Note: This course was originally designed and developed by [Tom Donoghue](#). While lectures, assignments, exams and coding labs will be altered from the original run of the course in Fall 2018, tons of credit for this course is due to Tom for his awesome work getting this course off the ground.

The PDF slides from the start of the first class are available here: https://cogs18.github.io/assets/intro/01_welcome.pdf



If you have not done any MATLAB or Python (or any) programming, or is feeling rusty with Python, please go to this course and complete up to (including) Lecture 11.

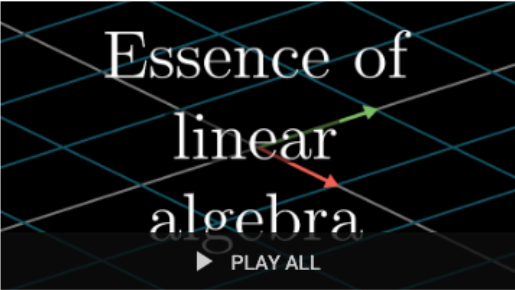
<https://cogs18.github.io/materials/00-Introduction/>

Download the notebooks, and try to answer the clicker questions before running the code to confirm your answers, and **experiment!**



1. Get (re)acquainted with GitHub & python
2. Vectors and dot product
3. Complex numbers





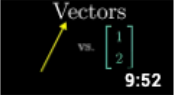
Essence of linear algebra


15 videos • 5,962,307 views • Last updated on 16 Mar 2019


A geometric understanding of matrices, determinants, eigen-stuffs and more.


3Blue1Brown SUBSCRIBED 1.8M


SEASON 1 ▾


- 

3BLUE1BROWN SERIES S1 • E1
Vectors, what even are they? | Essence of linear algebra, chapter 1
3Blue1Brown
- 

3BLUE1BROWN SERIES S1 • E2
Linear combinations, span, and basis vectors | Essence of linear algebra, chapter 2
3Blue1Brown
- 

3BLUE1BROWN SERIES S1 • E3
Linear transformations and matrices | Essence of linear algebra, chapter 3
3Blue1Brown
- 

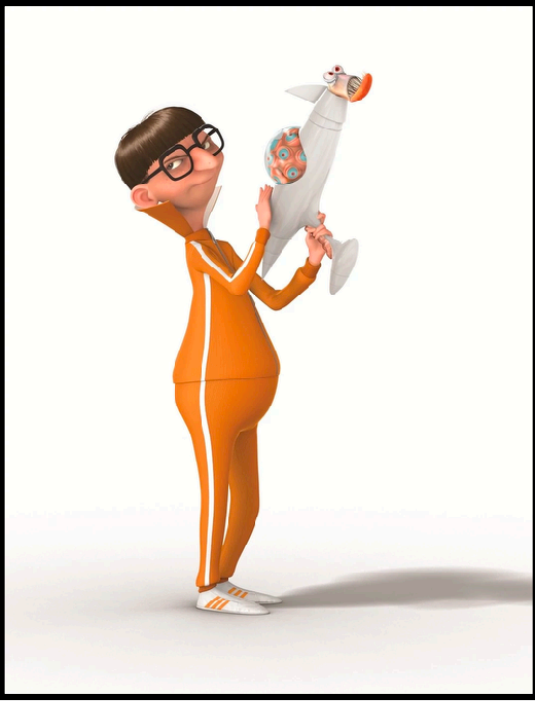
3BLUE1BROWN SERIES S1 • E4
Matrix multiplication as composition | Essence of linear algebra, chapter 4
3Blue1Brown
- 

3BLUE1BROWN SERIES S1 • E5
Three-dimensional linear transformations | Essence of linear algebra, chapter 5
3Blue1Brown
- 

3BLUE1BROWN SERIES S1 • E6
The determinant | Essence of linear algebra, chapter 6
3Blue1Brown

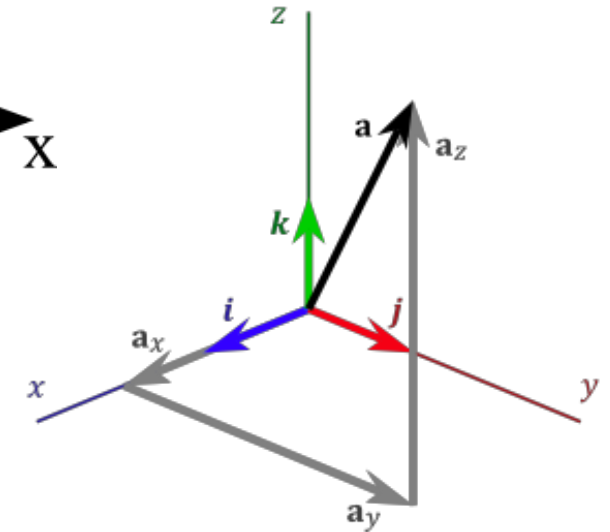
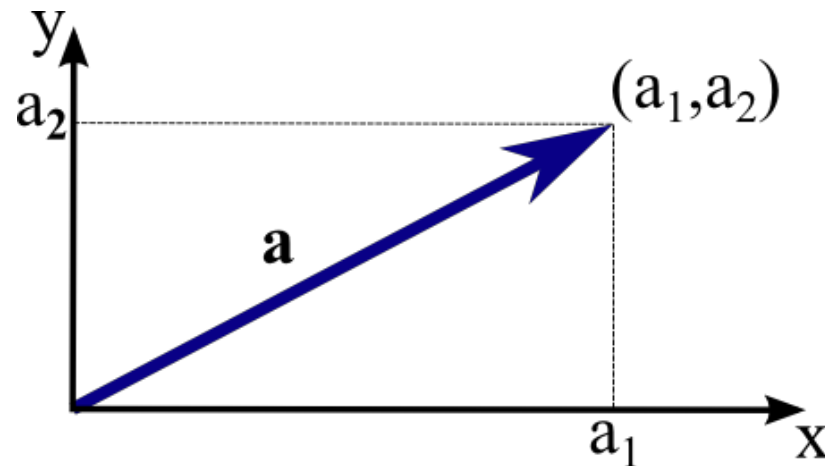


Vectors & Arrays



A vector is most intuitively thought about graphically.
It has a **direction** and a **magnitude**.

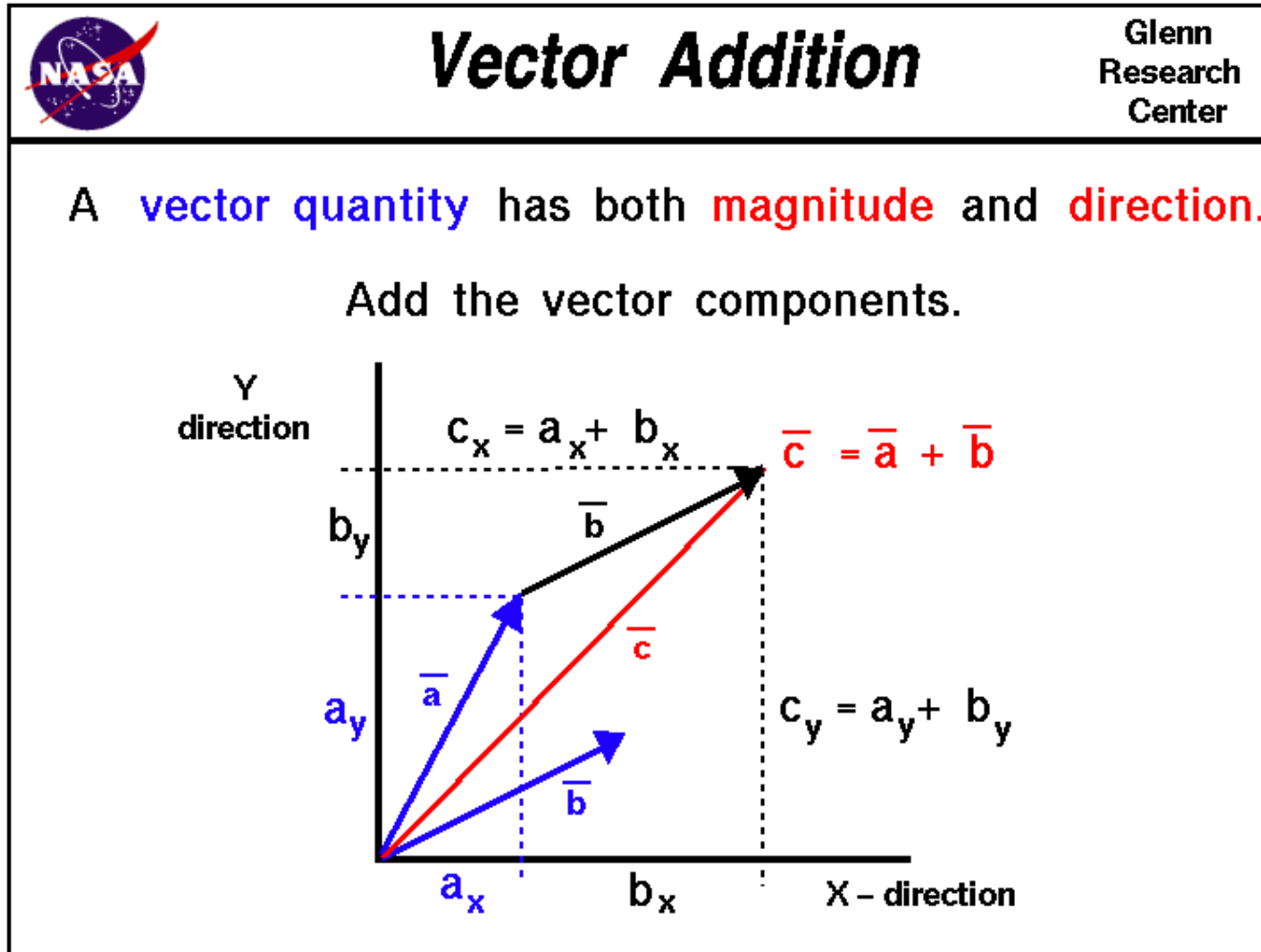
Can be in arbitrary number of dimensions.



Conveniently, in Python (and most other languages),
an **N-dimensional vector** can be represented as a
length-N array.



Operations between vectors (adding, multiplying, etc) are done element-wise.



Algebraic definition [\[edit \]](#)

The dot product of two vectors $\mathbf{a} = [a_1, a_2, \dots, a_n]$ and $\mathbf{b} = [b_1, b_2, \dots, b_n]$ is defined as:^[1]

$$\mathbf{a} \cdot \mathbf{b} = \sum_{i=1}^n a_i b_i = a_1 b_1 + a_2 b_2 + \dots + a_n b_n$$

where Σ denotes [summation](#) and n is the dimension of the [vector space](#). For instance, in [three-dimensional space](#), the dot product of vectors $[1, 3, -5]$ and $[4, -2, -1]$ is:

$$\begin{aligned} [1, 3, -5] \cdot [4, -2, -1] &= (1 \times 4) + (3 \times -2) + (-5 \times -1) \\ &= 4 - 6 + 5 \\ &= 3 \end{aligned}$$

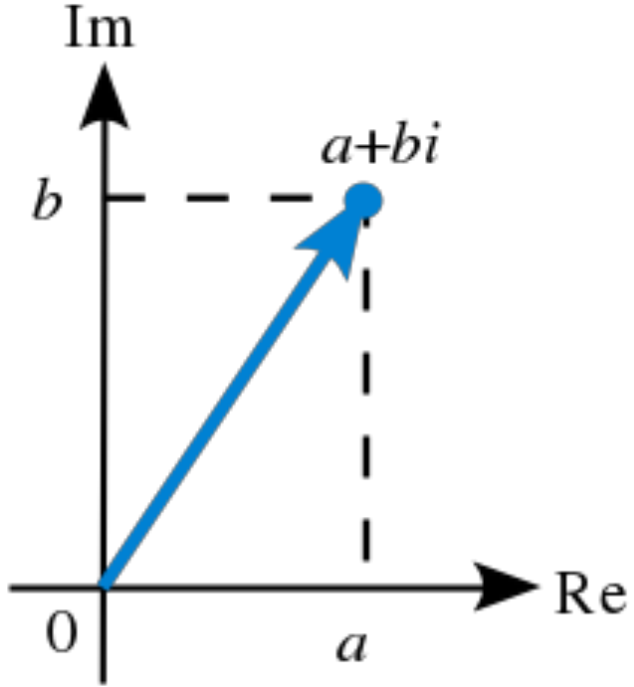
We'll be using various forms of this. **A LOT.**



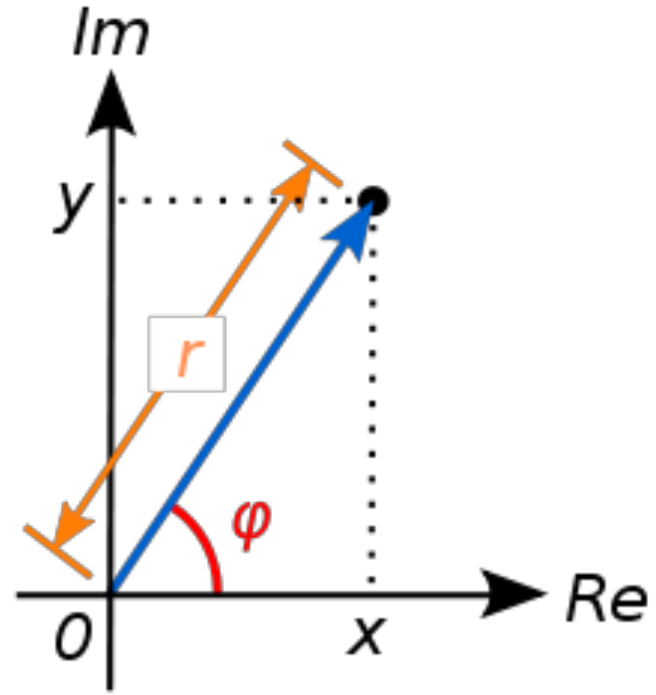
1. Get (re)acquainted with GitHub & python
2. Vectors and dot product
3. Complex numbers



Complex Numbers



Rectangular Form



Polar Form

