# █ Managing Multidimensional Arrays

## ▓▓▓ An xarray tutorial

## Introduction

### Janukan Sivajeyan, PhD student @ UAlberta

- Reducing Heat Generated By High Intensity Lasers
- Simulations with TBs of array data

### Materials

- will be on github [code/presentation]

## Arrays and Dimensions

An array is an ordered grouping of values (often numeric)

- "0D": Single value a.k.a. **scalar**
- 1D: Line of Values a.k.a **vector**
- 2D: Rectangle of Values a.k.a **matrix**
- 3D: Rectangular Prism of Values
- 4D: Rectangular Prism Movie of Values

## Shape of Array

- An array's **shape** is the length of each dimension
- **shape** *can* be expressed as a 1D array
- [2,4,3] could be the **shape** of a 3D array

## Indices [Plural of Index]

- Positions within the array are addressed by **indices**
- **Indices**: ordered grouping of integers [1 per dim.]
- **Indices** *can* be expressed as a 1D array

### Example Array

```
# Array Values
[9, 4, 3, 5, 4]
# Array Indices
[0, 1, 2, 3, 4]
```

## ■ Indices: 2D Arrays

### ▨ Values

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

## Indices: 2D Arrays

- The 2D arrays have 2 element **indices**

### Position [Index]

```
[[(0,0), (0,1), (0,2)],
 [(1,0), (1,1), (1,2)],
 [(2,0), (2,1), (2,2)]]
```

## Coordinates

- Array Indices may map to a **coordinate** (x,y,z,t)
- Aligned: **coordinate** element vary along only one dim.
- Aligned **coordinates** can be represented with a 1D array
- Does not require a regular interval

## Coordinates Example

### Example Array

```
# Values
[9, 4, 3, 5, 4]
# Indices
[0, 1, 2, 3, 4]
# Coordinate
[0.1, 0.2, 0.4, 0.5, 0.6]
```

## ■ Coordinates vs Indices

- Arrays always have **indices**
- **Indices** are always integer values
- **Coords** may be integer values
- Aligned coords have coordinate vectors

## ▨ Graph on Grid Paper Analogy

The grid lines represent indices and the xticks/yticks represent the coordinate vectors

## █ Applications

- Linear Algebra
- Quantities at various points in space and time
- Geospatial Data, Simulation Data
- Neural Network Weights [2D]
- Media [Music, Photos, Videos]

## ▓▓ Programming

- Languages: MATLAB, Fortran, J, APL
- Libraries: numpy, dask, pytorch

## Indexing Issues

Common indexing issues with arrays

- Mapping coordinates to indexes and vice-versa
- Order of coordinates: (x,y,z) vs. (x,z,y)

### Example

I have a 2D array of temperatures [temp] and I want to find the values at Norquest

```
temp[25,46]
```

## Indexing Issues

```
temp[25,46]
```

### 25 and 46 are indices

- the indices maps to physical coordinates
- numpy handles indices not coordinates
- "manual" translation between coords. and indices

## Indexing Issues

```
temp[25,46]
```

### Which one is which?

- one dimension is longitude, other is latitude
- in numpy, you have to "keep track" of order
- flipping indices will cause issues [wrong/invalid]

## Indexing Issues
## xarray

Python library for managing arrays with 2 main features

- **Labeling**: Combine arrays with coordinate vectors
- **Ecosystem**: Integrates with other array libraries

Reduce some of the "manual effort" with array programming

**Today's talk only covers a small part of xarray**

## Image Arrays

Images can be represented as 3D arrays. Here is one way to describe an image.

### Dimensions

- Length
- Width
- Colour: Red, Green, Blue

### Values

Array values are floats between 0 and 1

## Procedural Image Generation

Creating randomly generated images

1. Low Resolution Random Noise
2. "Smooth" the noise
3. Image manipulation [Math on arrays]

## Questions?