

# MATLAB

## 3-dimensional data

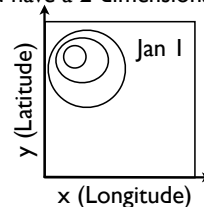
Tuesday, 1 October 13

1

## 3-dimensional data

Gridded satellite data for the oceans are 3-dimensional.

For a given date, you have a 2-dimensional map in x and y.



Tuesday, 1 October 13

2

## 3-D data

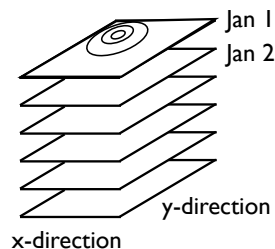
The three dimensions are the x-direction, y-direction, and with time in the 3rd dimension.

Tuesday, 1 October 13

3

## 3-dimensional data

In Excel, 3-d data could be represented by many sheets, one for each date.



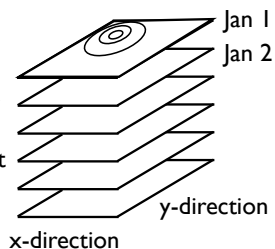
Tuesday, 1 October 13

4

## 3-dimensional data

In Matlab, they are still given by a *variable*, e.g. `sst1`, but it is now 3-D.

Just as for 1-D time series, a separate vector must define the axes (e.g. lon, lat and time)

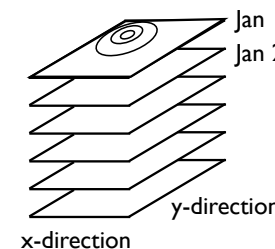


Tuesday, 1 October 13

5

## Another way to think about 3-d data

For a given location, you have a 1-dimensional data series, varying in time (time series).



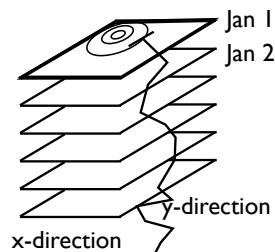
Tuesday, 1 October 13

6

## Another way to think about 3-d data

Now, instead of thinking of the 3-d data as a stack of paper, where each paper is one map,

... it is like a bundle of strings where each string is one time series.

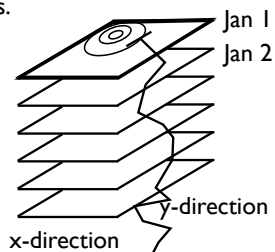


Tuesday, 1 October 13

7

## Another way to think about 3-d data

In Excel, this is like taking the data from a single cell, e.g. A1 or C3, but for each of the many sheets.



Tuesday, 1 October 13

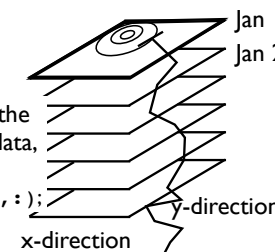
8

## Another way to think about 3-d data

In Matlab, it is finding the pixel in x- and y-space which is associated with a particular location.

...then getting the time variable data, e.g.

```
sst1(13,200,:);
```



Tuesday, 1 October 13

9

## Extracting data: recall indices

You can access a particular value in the 2-d matrix using the row and column index.

DIM: 2 →

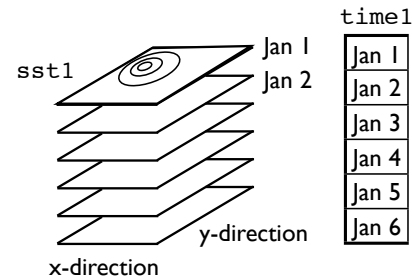
|     |     |     |     |     |
|-----|-----|-----|-----|-----|
| DIM | 10  | 20  | 30  | 40  |
| ↓   | 110 | 120 | 130 | 140 |
| ↓   | 210 | 220 | 230 | 240 |

Tuesday, 1 October 13

10

## Extracting a map

To get a single map out of a 3-D matrix, you need to specify the time index.

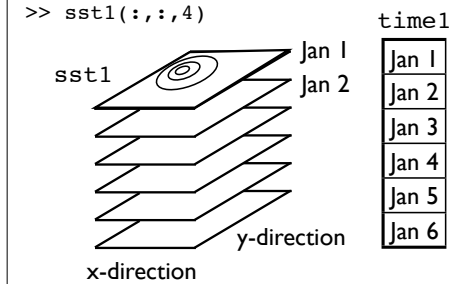


Tuesday, 1 October 13

11

## Extracting a map

So to get to the map from Jan 4, where `time1(4)` is Jan 4, you would use

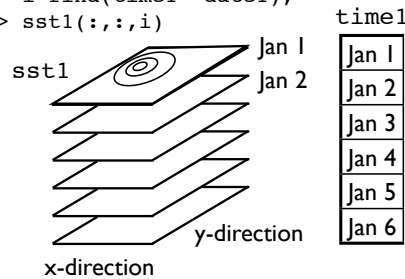


Tuesday, 1 October 13

12

For SST on any date, you (1) find the row in `time1` containing that date, suppose it's the  $i^{\text{th}}$  row; (2) then use the  $i$ -index in the 3rd dimension in `sst1`.

```
>> date1=datenum(2000,1,3);
>> i=find(time1==date1);
>> sst1(:,:,i)
```

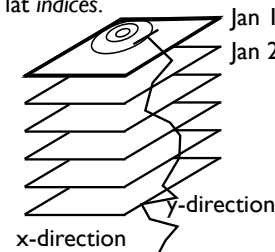


Tuesday, 1 October 13

13

## Extracting a time series

To get a time series out of a 3-D matrix varying in lon, lat and time, you need to specify lon and lat indices.

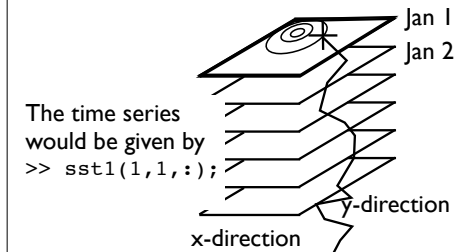


Tuesday, 1 October 13

14

## Extracting a time series

Suppose the + corresponds to the first row and column (A) in each sheet.



Tuesday, 1 October 13

15

## Extracting a time series using *find*

To extract an SST time series for a general location, you (1) find the index in the lon variable containing that longitude, and (2) the index for the lat, call them the  $j^{\text{th}}$  column and  $i^{\text{th}}$  row; (3) then use the  $i$ -index in the 1st dimension in `sst1` and  $j$  in the 2<sup>nd</sup>.

```
>> i=find(lat==26);
>> j=find(lon==-76);
>> sst1(i,j,:);
```

Tuesday, 1 October 13

16

## Review

1. Satellite data are commonly 3-dimensional surface map-time series, where the dimensions are latitude, longitude and time.

2. 2-dimensional maps, or 1-dimensional time series can be extracted, using the auxiliary time, lat and/or lon vectors.

Tuesday, 1 October 13

17