Spatial Data and Analysis

Discussion 1

Felipe González

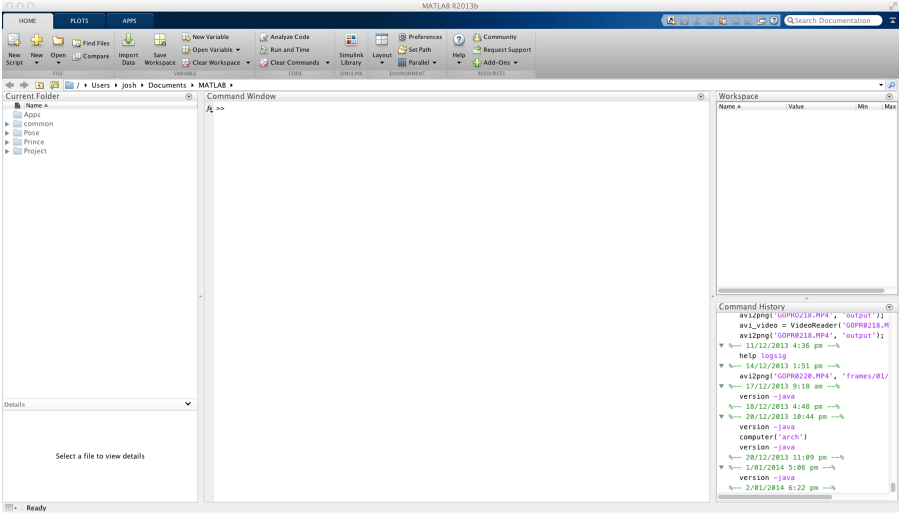
UC Berkeley

August $28th$

Outline

1. Basics
2. Matrices
3. Figures
4. M-Files
5. Exercises

Basics — MATLAB



Basics — windows

1. Command Window: where calculations can be performed and numerical solutions are displayed. Example: calculation `2+2` causes `ans = 4` to be displayed

2. Workspace: window in which imported data and variables created by you are stored. Example: definition `A = 4` causes variable `A` to be stored

3. Current Folder: where current project is located

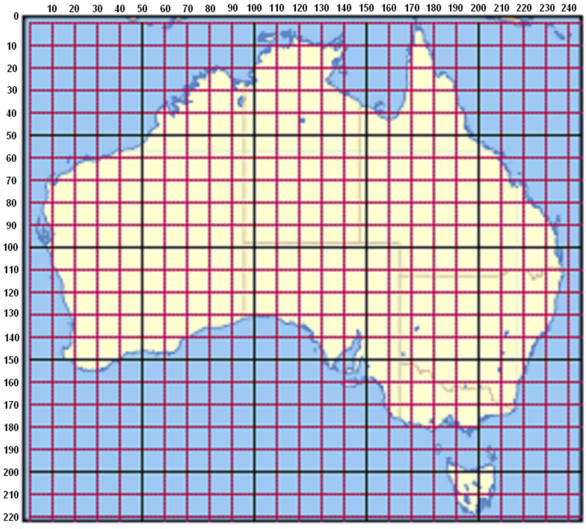
4. Command History: previously used commands

Basics — useful commands

`clear`	:	deletes all current data
`clc`	:	deletes everything in the command window
`cd('DIR')`	:	sets current directory
`+ -`	:	addition and subtraction
`* /`	:	multiplication and division
`^`	:	exponent
`Inf`	:	infinity
`pi`	:	π (number)
`NaN`	:	not a number, equivalent to . in stata
`log(x) exp(x)`	:	natural logarithm and exponential of x
`help X`	:	information about command X
key ↑	:	previous commands

Matrices — why they matter

Matrices — why they matter



Matrices — why they matter

- Matrices help us to represent some variable x (e.g., rainfall) over space in a simple way:

$$\begin{bmatrix} x_{11} & x_{12} & x_{13} & \dots & x_{1n} \\ x_{21} & x_{22} & x_{23} & \dots & x_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{d1} & x_{d2} & x_{d3} & \dots & x_{dn} \end{bmatrix}$$

- Operations with these matrices representing space will give rise to "map algebra"

Over the course of five years, we geo-referenced and digitized the dates of each conflict (...) The geographic coverage of our sample is approximately from 8 *to* 78 *degrees latitude and from* -61 *to* 96 *degrees longitude. The region of interest is divided into equal-area square grids. The main results focus on 50×50km grids. Hence, the main sample contains 19,844 grids (...) When using 50×50km grids 11,210 of the cells are on land, with 1,783 on the European continent, 6,499 on the Asian continent, 2,638 on the African continent, and 290 on the North American continent.*

"Agricultural productivity, conflict and state size: evidence from potatoes, 1400–1900" (Iyigun, Nunn, and Qian 2015)


N
E
S
W
Legend
50 km grid cell
Copyright: ©2013 Esri, DeLorme, NAVTEQ

Matrices — definition

- MATLAB has been specially designed to work with matrices
- Defines $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$: `A = [1 2 ; 3 4]` or `A = [1,2;3,4]`
- Defines $A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$: `A = [1 2 3 ; 4 5 6 ; 7 8 9]`

Matrices — operations

Addition	:	`A + B`
Multiplication	:	`A * B`
Transpose	:	`A'`
Point-wise multiplication	:	`A.*B`
Point-wise exponentiation	:	`A.^B`
Inverse	:	`A^(-1)`
Right division	:	`A / B`
		`A * B^(-1)`
Left division	:	`A \ B`
		`A^(-1) * B`

Matrices — useful commands

`1:n`	:	$1 \times n$ vector
`(1:n)'`	:	$n \times 1$ vector
`zeros(n,m)`	:	$n \times m$ matrix of zeros
`ones(n,m)`	:	$n \times m$ matrix of ones
`eye(n,m)`	:	$n \times m$ identity matrix
`rand(n,m)`	:	$n \times m$ matrix with uniformly distributed random numbers between 0 and 1
`randn(n,m)`	:	$n \times m$ matrix with normally distributed random numbers between 0 and 1
`size(A,x)`	:	size of matrix A ($x = 1$ rows, $x = 2$ columns)

Matrices — importing and exporting

- Data can be imported to MATLAB. The command to be used varies depending on the format of the data to be imported

`load 'X.mat'`	:	imports `'X'`
`X = csvread('X', R, C)`	:	imports `'X'`

- Data can also be exported. Same as before, command to be used depends on format...

`save 'X.mat'`	:	exports `'X'`
`dlmwrite('Y.csv', X)`	:	exports `'X'`

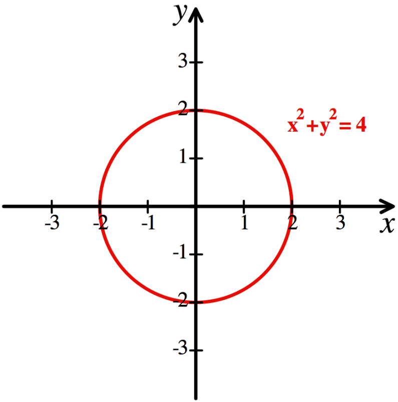
Coordinate systems

- You are asked to plot the following figure:

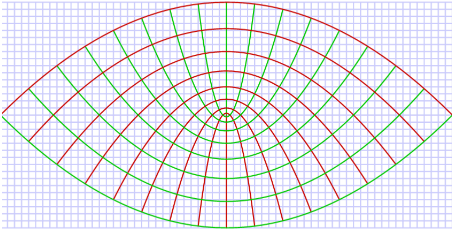
$$\text{Figure} \quad = \quad \begin{bmatrix} [1,1] \\ [1,3] \\ [2,2] \\ [2,1] \end{bmatrix}$$

- Easy to think about it in a Cartesian coordinate system where points correspond to (x, y) pairs, right? However, there are many other two-dimensional coordinate systems

Coordinate systems — theoretical examples

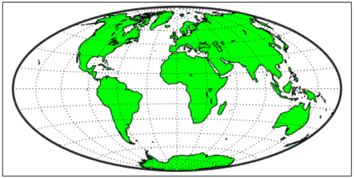


(a) Cartesian

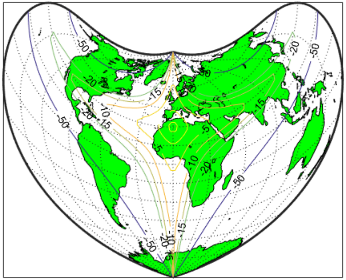


(b) Parabolic

Coordinate systems — empirical examples



(c) Hammer



(d) Bonne

Figures — useful commands

`plot(A)`	:	connected line with values of A $(n \times 1)$ in y-axis and row numbers in x-axis
`plot(A,B)`	:	connected line with values of A $(n \times 1)$ in x-axis and values of B $(n \times 1)$ in y-axis
`plot(A,B,'o')`	:	scatter plot with values of A $(n \times 1)$ in x-axis and values of B $(n \times 1)$ in y-axis
`help plot`	:	check for more options (e.g., color)

Figures — exporting

- To save a figure just write `saveas(gcf,'FIGURE.jpg')` after one of the previous commands. This will save the current figure as `.jpg` file in current directory
- If you do not specify the format it will assume the MATLAB figure file `.fig`
- Other available formats: `pdf`, `eps`, `png`, `tif`, and more

M-Files

- Collection of MATLAB commands
- How to create one:
    1. Choose `New` and `Script` from the `Home` menu
    2. Edit your file
    3. Choose `Save` from the `Editor` menu in the new window
- Execute (run) m-file using the 'Run' triangle button
- Solutions for the labs are a single or multiple m-files

Exercise — returns to education

- You are given the following data:

$$Y = \begin{bmatrix} 100 \\ 50 \\ 10 \\ 20 \\ 80 \\ 90 \\ 70 \end{bmatrix} \qquad X_1 = \begin{bmatrix} 20 \\ 10 \\ 12 \\ 16 \\ 18 \\ 17 \\ 12 \end{bmatrix} \qquad X_2 = \begin{bmatrix} 36 \\ 45 \\ 38 \\ 50 \\ 42 \\ 52 \\ 37 \end{bmatrix}$$

- Estimate the model $Y = \alpha + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_2^2$ by OLS

Exercise — returns to education

```
% Clears memory and screen
    clear ; clc

% Creates matrices
    Y  = [100 50 10 20 80 90 70]' ;
    X1 = [20  10 12 16 18 17 12]' ;
    X2 = [36  45 38 50 42 52 37]' ;

% Estimates OLS coefficients
    X  = [ones(size(Y,1),1) X1 X2 X2.*X2] ;
    b  = (X' * X) \ (X' * log(Y))         ;
```

Exercise — random cities

- Generate 1,000 random locations from a normal distribution. Make the center of this 'random city' to be GSPP, and 95% of locations to be located within 4 kilometers
- Create a map with locations in blue and the center in red
- Repeat previous exercise using draws from a uniform distribution
- What are the differences between the 'cities' we just simulated?

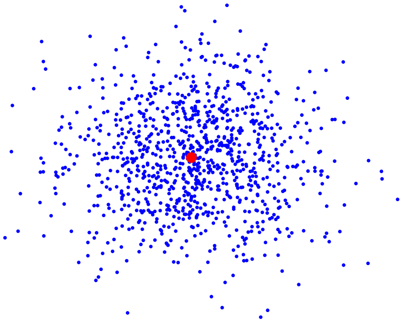
Normal city

```
% Center
    center_lat =   37.875999 ;
    center_lon = -122.257834 ;

% Locations from normal distribution
    lat = center_lat + 2*randn(1000,1);
    lon = center_lon + 2*randn(1000,1);

% Plots map
    hold on
    plot(lon,lat,'.b')
    plot(center_lon,center_lat,'.r','MarkerSize',30)
    hold off
```

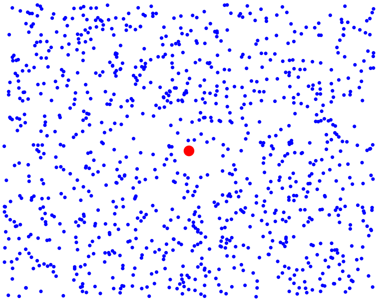
Uniform city

```
% Center
    center_lat =   37.875999 ;
    center_lon = -122.257834 ;

% Locations from uniform distribution
    lat = center_lat + 2*rand(1000,1) - 1;
    lon = center_lon + 2*rand(1000,1) - 1;

% Plots map
    hold on
    plot(lon,lat,'.b')
    plot(center_lon,center_lat,'.r','MarkerSize',30)
    hold off
```

Two random cities



(a) Normal



(b) Uniform

Additional resources

1. If you're submitting your lab in TeX check out the `mcode` package. Google 'mcode', and put the file `mcode.sty` in the folder. with your .tex file. You can also find this package in the bCourse for this class: `Files/Resources`

2. You can start doing some research on the `export_fig` package for MATLAB to export figures nicely in vector and bitmap formats. We'll review this in following sections

3. As I will not be here next week, I'll have additional office hours September 8*th* from 4-6pm (Quiet Study Room)