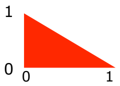
Patches of Color

Outline

- Announcements
  - Homework I due TODAY. 5PM by e-mail
  - Homework II on web
  - No lecture on Monday (Fall Break)
  - HW II due next Friday, HW III out then
- Patches in 2D--pcolor
- Example: NWtopex
- Survey

Drawing patches

- Lots of functions produce patches
  - bar, fill, fill3
- Patches are most flexible 2D objects
- patch is the lowest level function (followed closely by fill)
  - patch(x,y,c)--x and y specify vertex coordinates, c controls the color
    - C=color character (‘r’,’b’,’g’,etc.)
    - RGB vector
    - “color index”

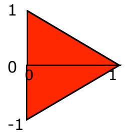


x=[0 1 0]
y=[0 0 1]
c=‘r’

Drawing patches

- patch(X,Y,C)--Each column of X, Y, and C is a different polygon,
  - but same object!
  - X and Y must be the same size
    - Each polygon must have same number of vertices (rows)

```
X=[0 1 0;
      0 1 0]’
Y=[0 0 1;
      0 0 -1]’
C=‘r’
```

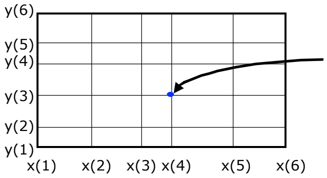


Key properties of patch objects

- edgecolor--color of the edges
- facecolor--color inside the the patch
- Both of these can be set to a specific color (or none)
- Or, we can prescribe another dimension of data at each vertex and let it control the color

Patches and data

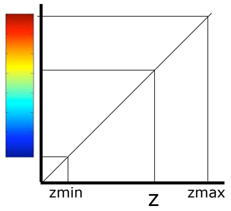
- We’re moving from y=f(x) to z=f(x,y)
- Typically, x & y are defined on a grid of points:



data lives at points (nodes) Z(3,4)=f(x(4),y (3))

Colorizing z

- A standard way of representing 2D data is to make color indicate z

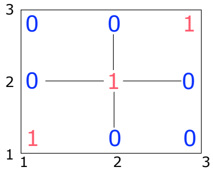


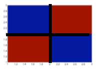
pcolor

- pcolor(x,y,Z) will colorize Z on grid defined by x and y
  - Z=m-by-n, x=1-by-n, y=m-by-1
- pcolor(X,Y,Z) will colorize Z on an irregular grid
  - X,Y, and Z all m-by-n
- h=pcolor(…) gets the handle.
  - The object is actually a surface object
  - surface objects are nearly identical to patches, but must be constructed from quadrilaterals (a grid)

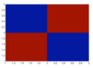
How it works

- h=pcolor(eye(3));

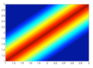




shading(‘faceted’)
color of cell is set by lower left-hand corner



shading(‘flat’)
edgecolor=‘none’



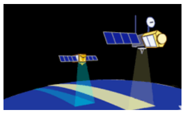
shading(‘interp’)
interpolates between vertices to get color

Controlling pcolor

- shading(str) sets ‘facecolor’ property to str
  - flat, faceted or interp
- colorbar shows a colorbar
- caxis([zmin, zmax]) controls the color limits
  - same as set(gca,’clim’,[zmin, zmax])
- colormap(cmap)--changes the colors. help graph3d lists the built in colormaps
  - we’ll learn how to “roll-your-own”

Example: NWtopex

- Since water flows down hill, sea-surface height (SSH) indicates currents
- The TOPEX/Poseidon satellite measures SSH with radar



NWtopex

- load(NWtopex)--loads NWtopex.mat which contains the following arrays:

name	size	description
lon	1-by-66	longitude (x)
lat	1-by-31	latitude (y)
SSH	31-by-66	SSH=z(x,y)
rkb	256-by-3	new colormap