Intro to Geographic Analysis

PAMELA TOMAN

Goals & agenda

Intro: How do we computationally represent geographic data?

Pro-tips / Gotchas:

- The best maps may still be flawed
- Use a spatial database
- Use hexagons to partition a surface
- Avoid the ecological fallacy & modifiable areal unit problem
- Do not analyze on lat-longs
- Divide out confounding variables
- Not all clusters are statistically significant
- Literal distance isn't always appropriate
- Spatial regression is tricky
- Consider network & other techniques
- Choropleths assume uniform potential
- Isopleths encourage assuming precision
- Cartograms distort shapes
- All colors are not equal

Conclusion: Let’s reflect on sample maps that visualize spatial data

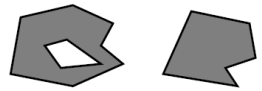
Points, lines and polygons are the basic unit of analysis

```
{
  "type": "Feature",
  "geometry": {
    "type": "Point",
    "coordinates":
[102.0, 0.5]
  },
  "properties": {
    "prop0": "value0"
  }
}
```

```
{
  "type": "Feature",
  "geometry": {
    "type": "LineString",
    "coordinates": [ [102.0, 0.0],
[103.0, 1.0], [104.0, 0.0],
[105.0, 1.0] ]
  },
  "properties": {
    "prop0": 0.0
  }
}
```

```
{
  "type": "Feature",
  "geometry": {
    "type": "Polygon",
    "coordinates": [ [ [100.0,
0.0], [101.0, 0.0], [101.0, 1.0],
[100.0, 1.0], [100.0, 0.0] ] ]
  },
  "properties": {
    "prop0": {"this": "that"}
  }
}
```

GeoJSON: A specification for representing geospatial objects



Substantial data exist as shapefiles

The “shapefile” format is a popular format for GIS (geographic information system) data; it is developed by Esri and interoperable with other GIS software

Shapefiles are an alternative to GeoJSON

Shapefiles, like GeoJSON, can be downloaded and bought

Mandatory files:
.shp – shape format (the feature geometry)
.shx – shape index format (index for query efficiency)
.dbf – attribute format

Other files are possible (e.g., projection, metadata, etc.)

Gotcha #1:
The best maps may still be flawed

Real-world edges are fractals

→ There is no boundary “truth”

Multiple sources exists for GIS data

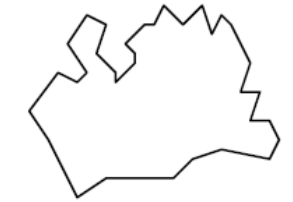
→ Overlaid layers may disagree

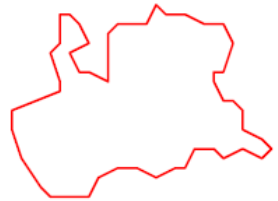
Digitization of maps is imperfect

→ Look for disconnected lines, switchbacks, loops, and other artifacts

Creating GIS data is expensive and irregularly done

→ Pay attention to creation date and source





(source)

Gotcha #2:
Use a spatial database

Databases use “indices” to speed up finding data

SSN:	123-45-6789	123-46-7890	124-46-7890	164-46-0000	518-99-1234	724-46-7890	918-22-4321
First:	Maria	Shaniqua	Maria	Tom	Ali	Ahmed	Pham
Last:	Smith	Lewis	Lopez	Jones	Lewis	Shehzad	Nguyen

Find person with SSN “164-46-0000”:

The index stores the data in sorted form.
This makes lookups fast.

How do we do fast lookups with an index?
We look at the middle of the data.
- If it’s there, great.
- If not, look to the middle of the direction it will be in.
- Repeat.

Each step rules out ½ the data.

Gotcha #2:
Use a spatial database

Databases use “indices” to speed up finding data

SSN:	123-45-6789	123-46-7890	124-46-7890	164-46-0000	518-99-1234	724-46-7890	918-22-4321
First:	Maria	Shaniqua	Maria	Tom	Ali	Ahmed	Pham
Last:	Smith	Lewis	Lopez	Jones	Lewis	Shehzad	Nguyen

Find person with SSN “164-46-0000”:

The index stores the data in sorted form.
This makes lookups fast.

How do we do fast lookups with an index?
We look at the middle of the data.
- If it’s there, great.
- If not, look to the middle of the direction it will be in.
- Repeat.

Each step rules out ½ the data.

Gotcha #2:
Use a spatial database

Databases use “indices” to speed up finding data

SSN:	123-45-6789	123-46-7890	124-46-7890	164-46-0000	518-99-1234	724-46-7890	918-22-4321
First:	Maria	Shaniqua	Maria	Tom	Ali	Ahmed	Pham
Last:	Smith	Lewis	Lopez	Jones	Lewis	Shehzad	Nguyen

Find person with SSN “164-46-0000”:

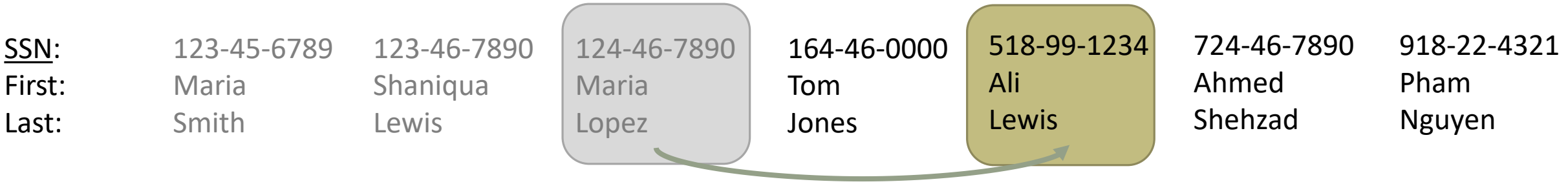
The index stores the data in sorted form.
This makes lookups fast.

How do we do fast lookups with an index?
We look at the middle of the data.
- If it’s there, great.
- If not, look to the middle of the direction it will be in.
- Repeat.

Each step rules out ½ the data.

Gotcha #2:
Use a spatial database

Databases use “indices” to speed up finding data



Find person with SSN “164-46-0000”:

The index stores the data in sorted form.
This makes lookups fast.

How do we do fast lookups with an index?
We look at the middle of the data.
- If it’s there, great.
- If not, look to the middle of the direction it will be in.
- Repeat.

Each step rules out ½ the data.

Gotcha #2:
Use a spatial database

Databases use “indices” to speed up finding data

SSN:	123-45-6789	123-46-7890	124-46-7890	164-46-0000	518-99-1234	724-46-7890	918-22-4321
First:	Maria	Shaniqua	Maria	Tom	Ali	Ahmed	Pham
Last:	Smith	Lewis	Lopez	Jones	Lewis	Shehzad	Nguyen

Find person with SSN “164-46-0000”:

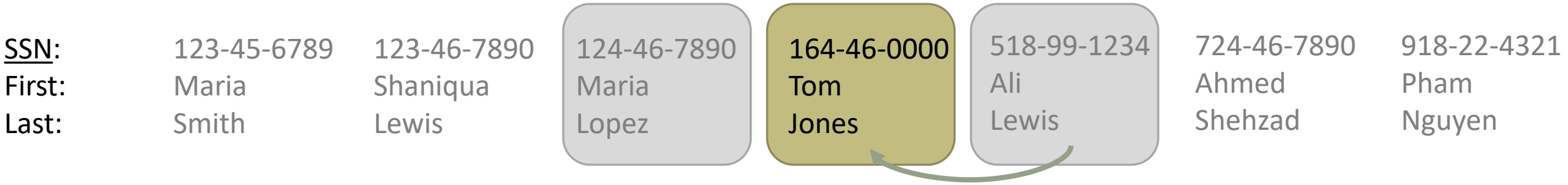
The index stores the data in sorted form.
This makes lookups fast.

How do we do fast lookups with an index?
We look at the middle of the data.
- If it’s there, great.
- If not, look to the middle of the direction it will be in.
- Repeat.

Each step rules out ½ the data.

Gotcha #2:
Use a spatial database

Databases use “indices” to speed up finding data



Find person with SSN “164-46-0000”:

The index stores the data in sorted form.
This makes lookups fast.

How do we do fast lookups with an index?
We look at the middle of the data.
- If it’s there, great.
- If not, look to the middle of the direction it will be in.
- Repeat.

Each step rules out ½ the data.

Gotcha #2:
Use a spatial database

Databases use “indices” to speed up finding data

SSN:	123-45-6789	123-46-7890	124-46-7890	164-46-0000	518-99-1234	724-46-7890	918-22-4321
First:	Maria	Shaniqua	Maria	Tom	Ali	Ahmed	Pham
Last:	Smith	Lewis	Lopez	Jones	Lewis	Shehzad	Nguyen

Find person with SSN “164-46-0000”:

The index stores the data in sorted form.
This makes lookups fast.

Instead of looking at every piece of data,
we only look at $\log_2(N)$.

How do we do fast lookups with an index?
We look at the middle of the data.
- If it’s there, great.
- If not, look to the middle of the direction it will be in.
- Repeat.

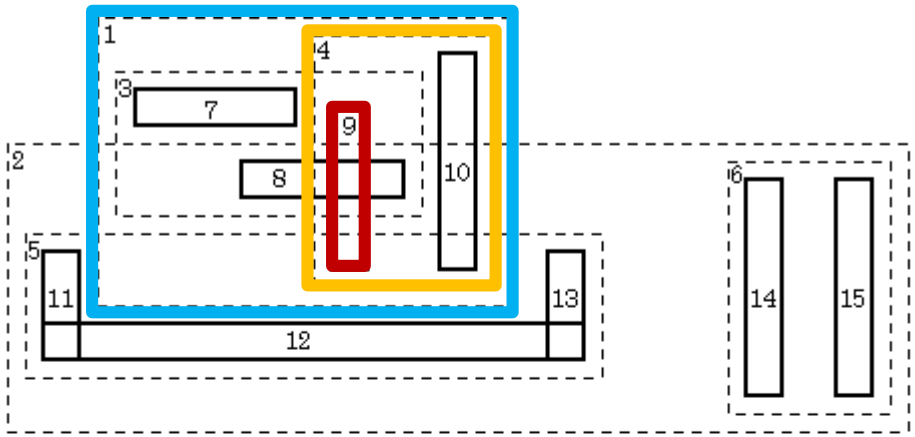
Each step rules out ½ the data.

Gotcha #2:
Use a spatial database

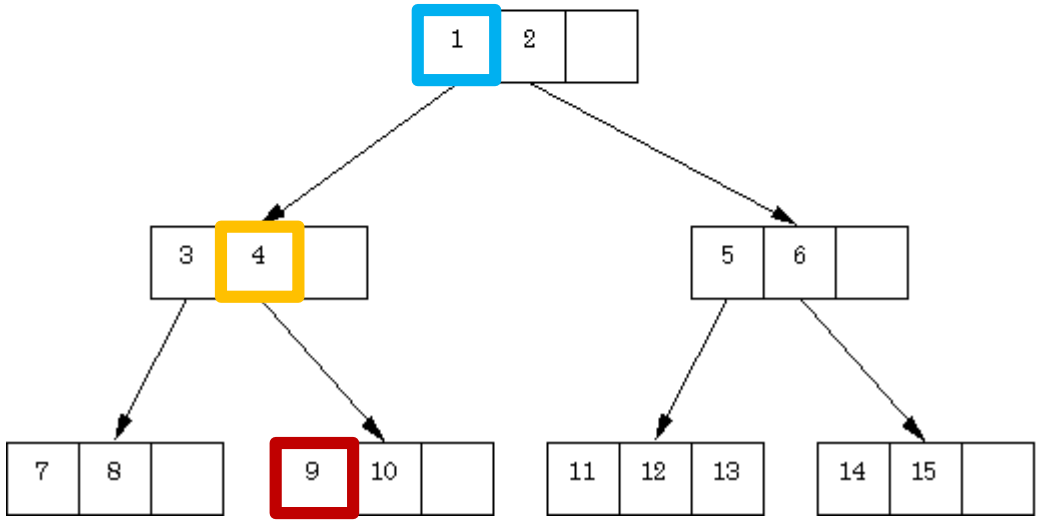
One way to index space is an **r-tree**

R-trees *coarsely approximate* the dataset:

- Nearby objects are grouped into a “minimum bounding rectangle”
- Each layer of the tree has smaller minimum bounding rectangles



Lookups can be fast:
Any query that doesn’t intersect at the top level cannot intersect any at any lower level



(source)

The Open Geospatial Consortium (OGC) has a standard

Usually geographic database support space through extensions:

- PostgreSQL → PostGIS
- Sqlite → SpatiaLite
- Oracle → Oracle Spatial

Gotcha #2:
Use a spatial database

One way to index space is an **r-tree**

R-trees *coarsely approximate* the dataset:

- Nearby objects are grouped into a “minimum bounding rectangle”
- Each layer of the tree has smaller minimum bounding rectangles

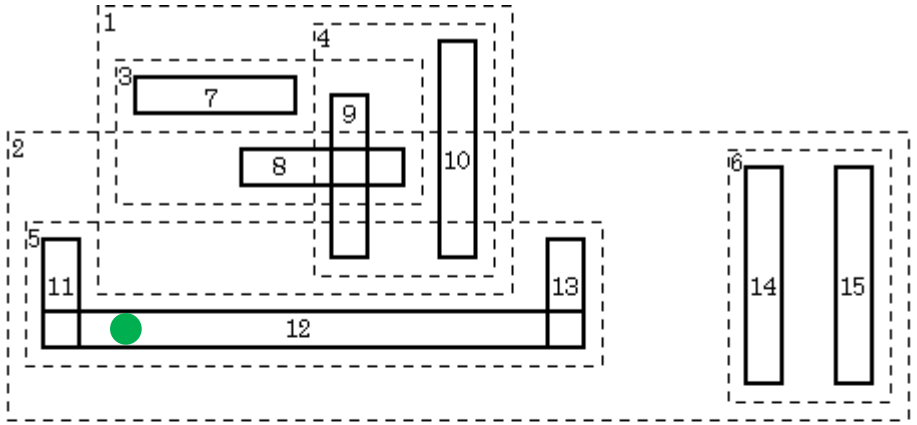
Lookups can be fast:
Any query that doesn’t intersect at the top level cannot intersect any at any lower level

The Open Geospatial Consortium (OGC) has a standard

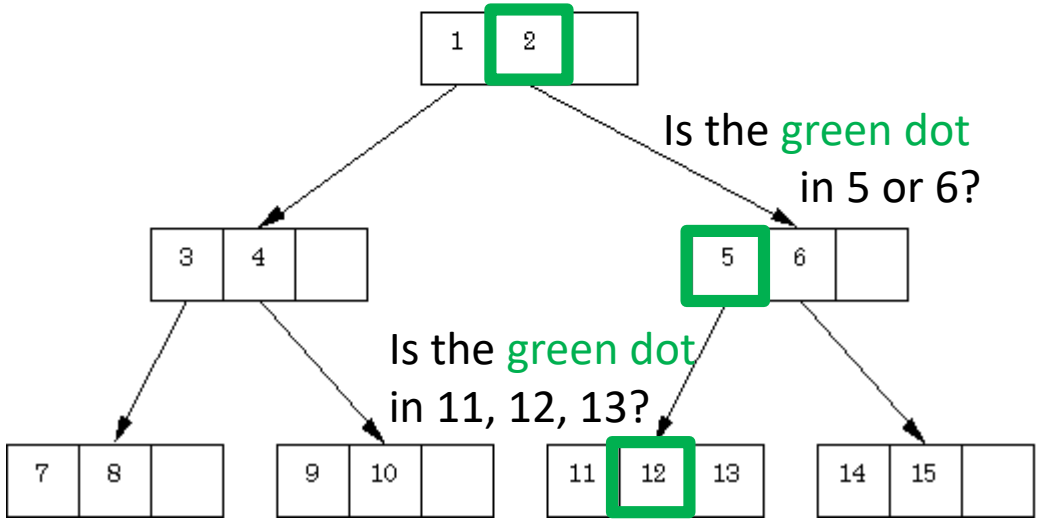
Usually geographic database support space through extensions:

- PostgreSQL → PostGIS
- Sqlite → SpatiaLite
- Oracle → Oracle Spatial

Each spatial area is a precinct of the city’s police force.
Which precinct manages the green dot?



Is the green dot in region 1 or 2?



Return the answer: the green dot is in precinct 12. (source)

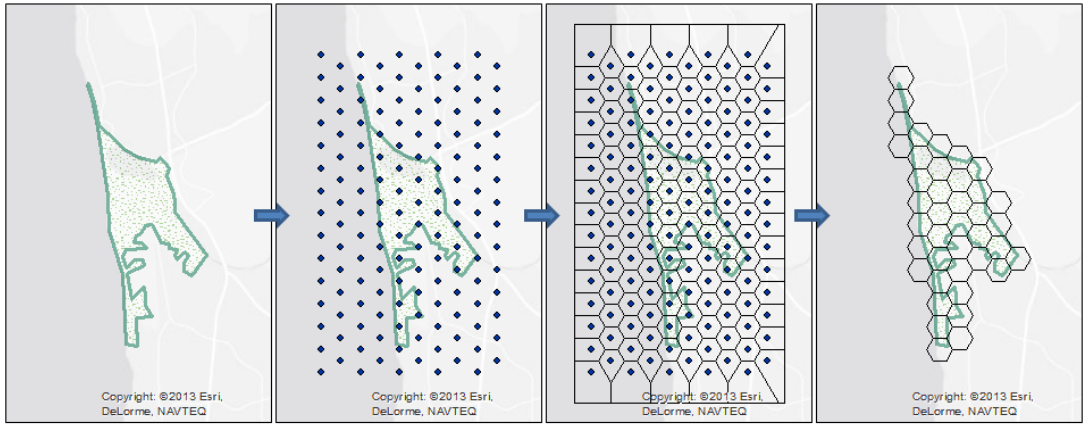
Gotcha #3:
Use hexagons to partition a surface

Hexagons are useful:

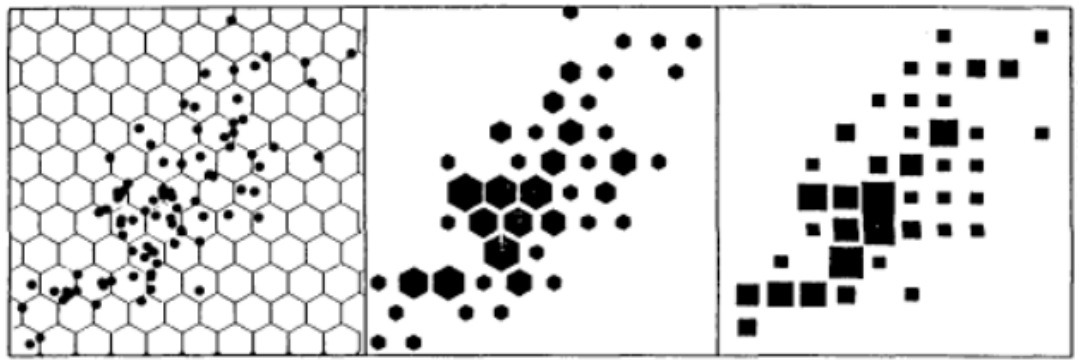
- Tessellate 2D surfaces
- Roundness ensures a coherent unit
- Produce a smooth & interpretable surface (no distracting lattice)
- Error due to data collection boundaries is visible
- Aesthetically pleasing

We use them in many ways:

- Visualization
- Smoothing point data into uniformly-sized bins for analysis
- Planning sampling to estimate density



(source)

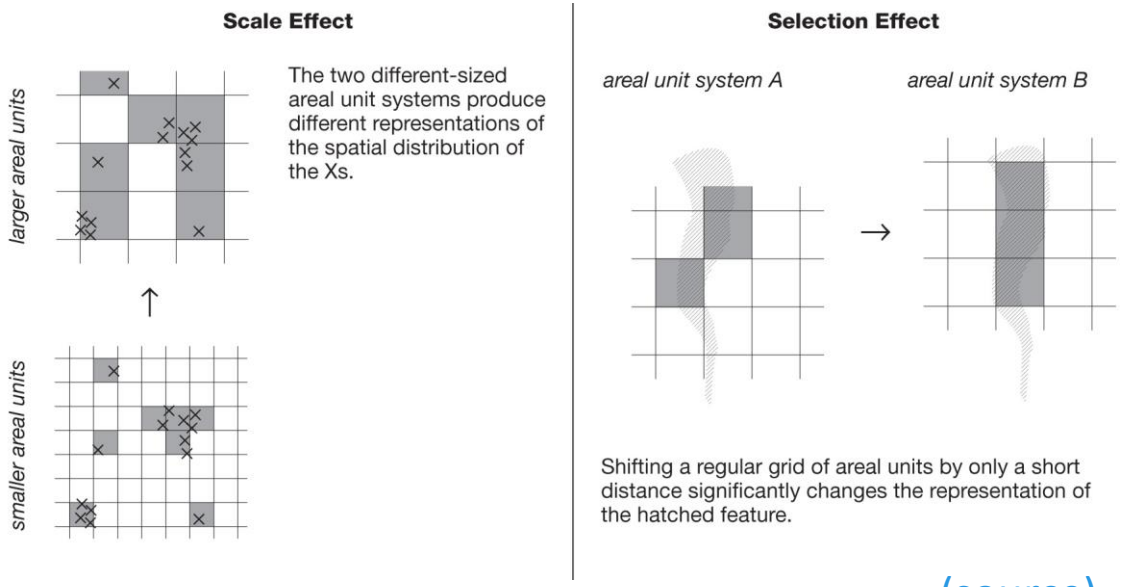


(source)

Gotcha #4:
Avoid the ecological fallacy & MAUP

Ecological fallacy: Can't draw conclusions about individuals given aggregated data

Modifiable areal unit problem: Conclusions depend on the "zones" used to aggregate spatial data (e.g., zip code vs. census tract). Simplest solution is to analyze the map at a smaller scale.



(source)

Gotcha #5:
Don’t analyze on lat-longs

Lat-longs are not a uniform unit of measure (1° latitude ≠ 1° longitude)

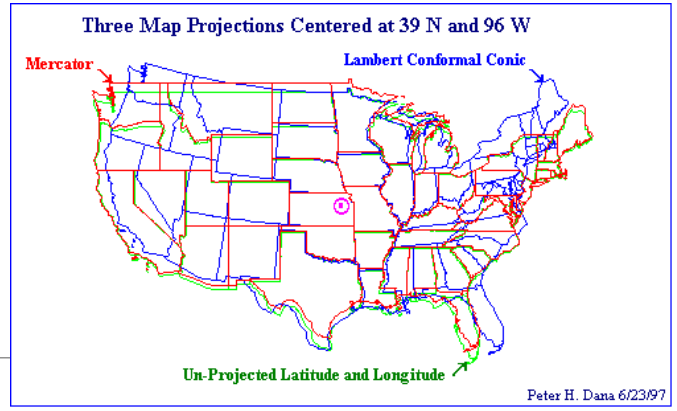
- Geographic Coordinate System (e.g., WGS 1983) or “datum”
- Refers to a 3D spherical surface
- Measured in degrees/minutes/seconds

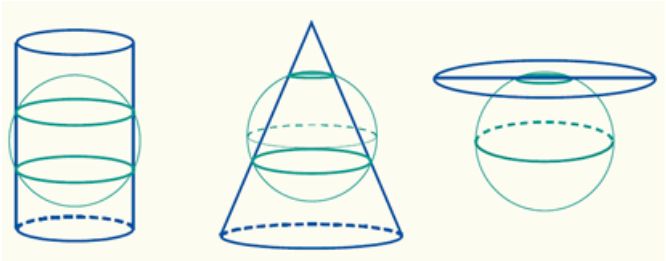
For analysis, always use a map projection (1 foot = 1 foot)

- Projected Coordinate System
  (based on a Geographic Coordinate System;
  can be cylindrical, conic, planar/azimuthal)
- Refers to a 2D flat surface
- Measured in feet/meters

Projections transform from a 3-D surface to a 2-D surface

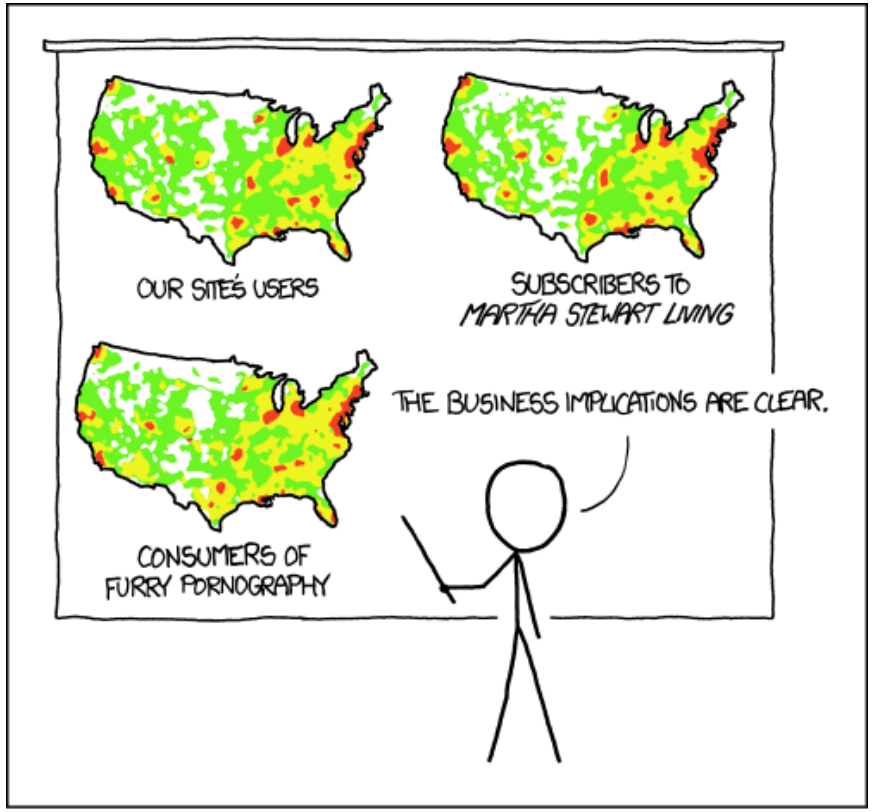
ArcGIS will project on the fly when it can, but that costs computation





(source)

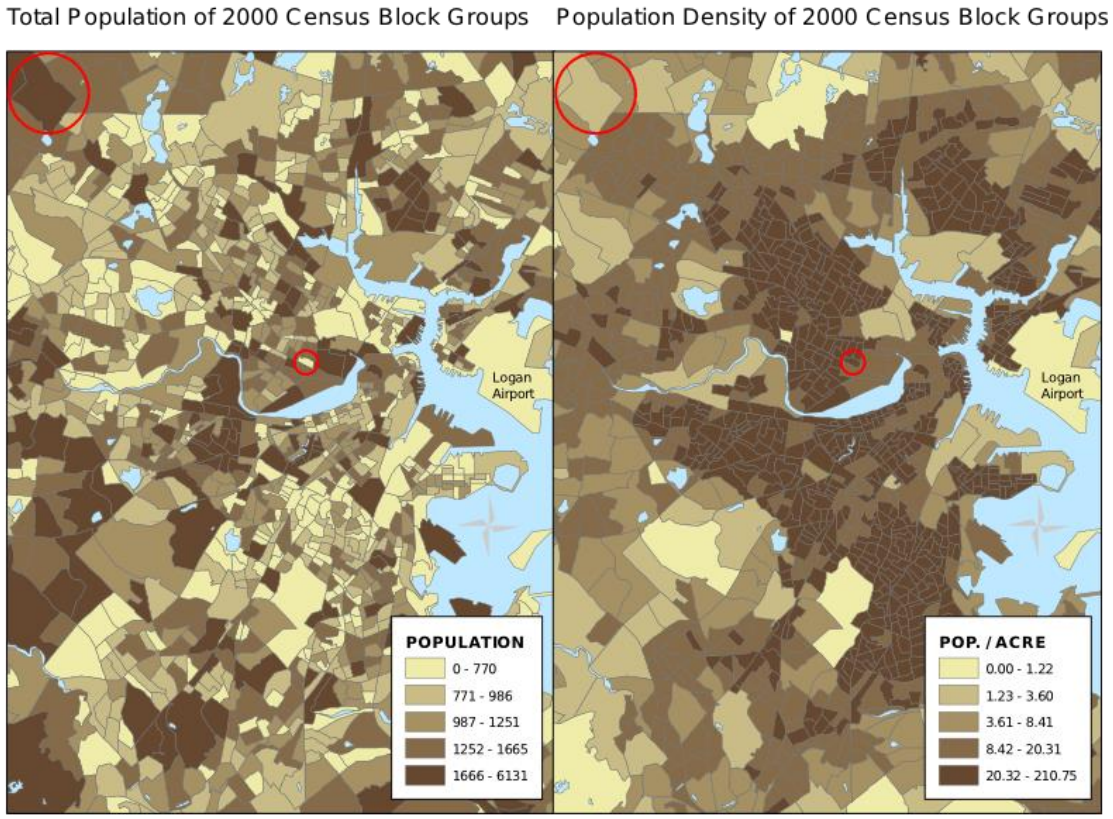
Gotcha #6:
Divide out confounding variables



PET PEEVE #208:
GEOGRAPHIC PROFILE MAPS WHICH ARE BASICALLY JUST POPULATION MAPS

(source)

Gotcha #6:
Divide out confounding variables



(source)

Population depends on area

Foreclosures depend on number of houses

Number of free and reduced lunches depends on population

…

Gotcha #7:
Not all clusters are statistically significant

In resource allocation problems, we want to know where a variable is especially high/low – with controls for random variation

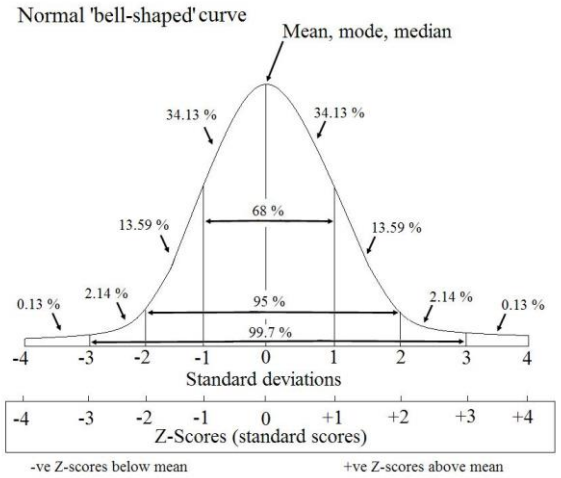
Hot spot analysis (also: spatial clustering, spatial auto-correlation) identifies statistically significant clusters in space:

- Getis-Ord Gi*
- Anselin Local Moran's I

Appropriate usage requires some thought:

- Unit of analysis (political units? hexagonal tessellation? something else?)
- Target variable (raw number? transformation of it?)
- Distance metric / scale of analysis

(source)



(source)

Gotcha #8:
Literal distance isn't always appropriate

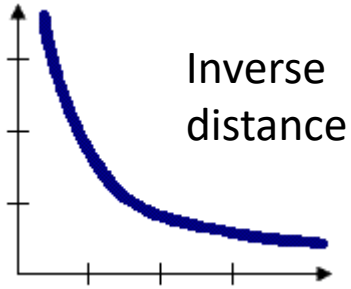
Most analyses consider spatial units relative to their neighbors

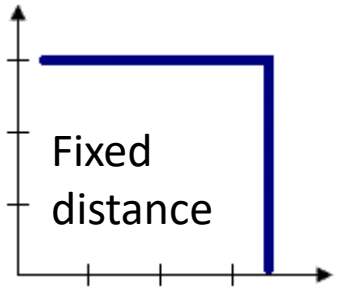
Defining "neighbors" appropriately is problem-specific

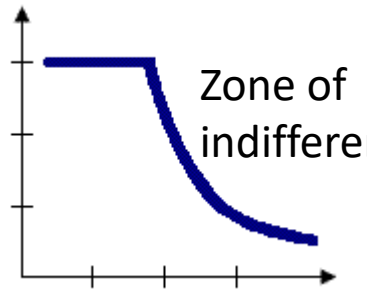
- **Nearness increases influence? → Inverse distance metric**
  Example: seed propagation
- **Sphere of indifference → Fixed distance metric**
  Good for hot spot analysis/binning
- **Local structure irrelevant, but drops beyond threshold → zone of indifference**
  Example: influence of job distance
- **Contextualize through neighbors → k-nearest neighbors**
  Ensures each feature is evaluated in context of neighbors, even when density is low

Sometimes non-physical distances matter more (e.g., time)

Topology and human geography matter:
Two spatially close neighborhoods may rarely interact if separated by highway





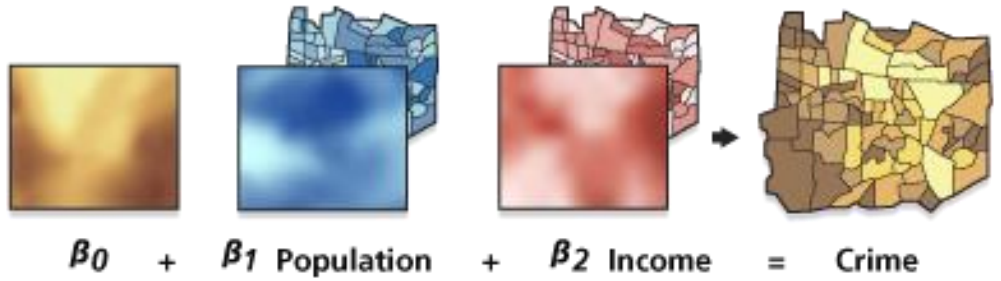


(source)

Gotcha #9:
Spatial regression is tricky

Spatial regression is useful:

- Make better policy decisions through increased understanding (modeling)
- Predict the future
- Test a hypothesis



(source)

Multiple methods enable Geographically Weighted Regression (GWR)

Figuring out which variables to include can be challenging

- Sometimes it's possible to identify missing variables through plotting the residuals (errors) looking for spatial explanations
  (e.g., overestimates on mountains and underestimate on valleys)
- But model specification & variable transformations are often tricky
- Regression is not a first level form of analysis

Gotcha #10:
Consider network & other techniques

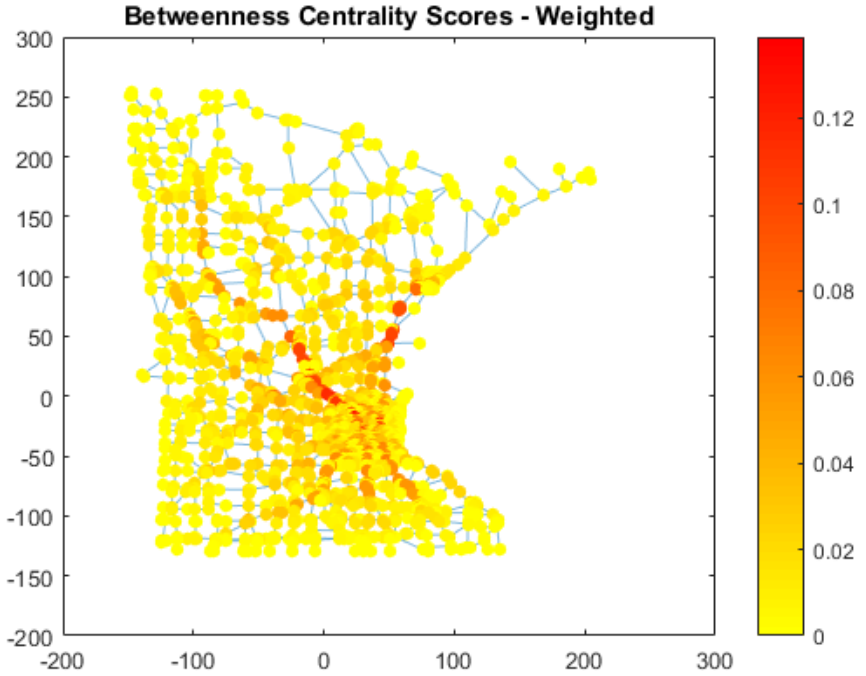
(source)

For some analytic questions, we might want to build new layers

Network analysis in particular can lend insight into interconnectedness (e.g., roads, utilities, rivers, …)

With network analysis, we can calculate:

- Relative importance of each juncture
- Highly interconnected clusters
- Shortest paths
- Sources & sinks
- Especially vulnerable junctures or links



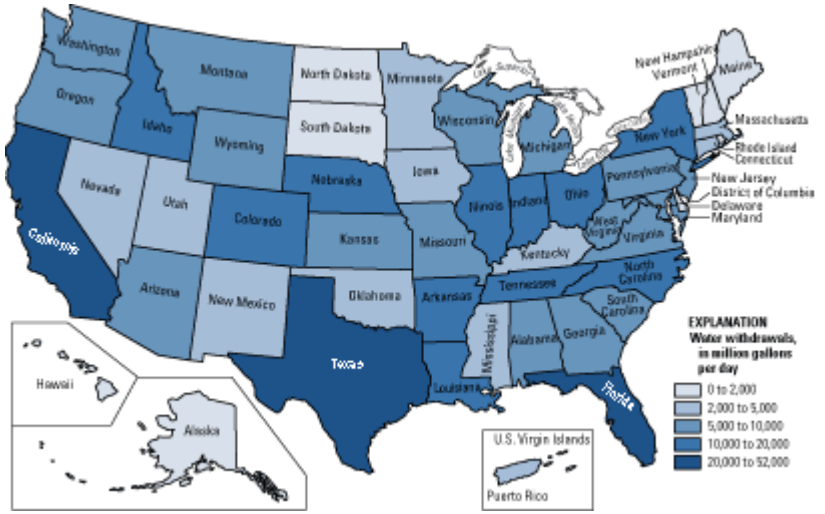
(source)

Gotcha #11:
Choropleths assume uniform potential

Choropleth maps use hue to show the value of a variable in defined geographic regions

Useful when the regions are important to a discussion (e.g., electoral map)

Assumes a relatively uniform distribution of the phenomenon in each region – so make sure this assumption is appropriate!



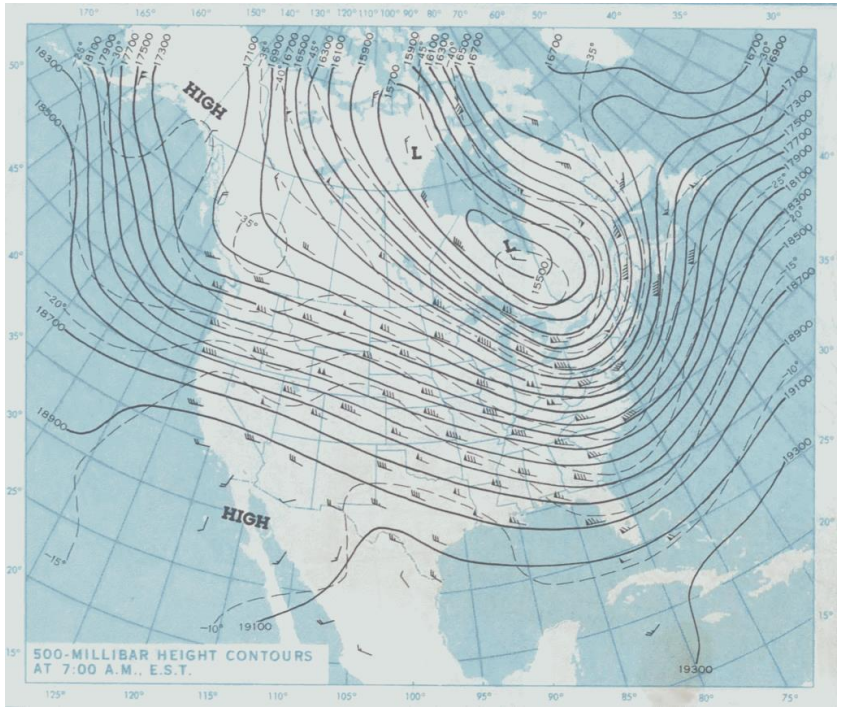
(source)

Gotcha #12:
Isopleths encourage assuming precision

Isopleth maps (also: isarithmic, contour) use lines to distinguish regions with the same value

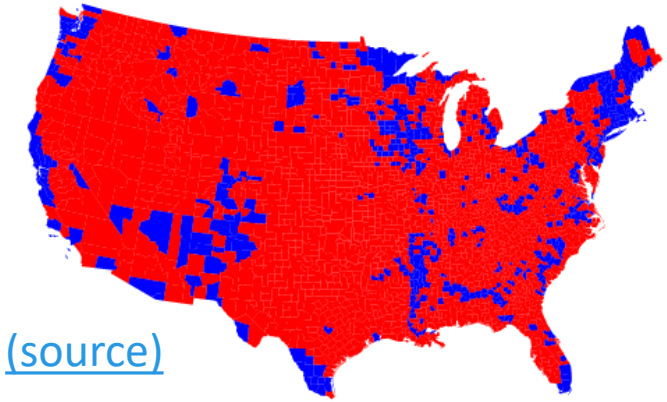
Data collection boundaries can arbitrarily change the line, and the user won't know the effect

People assume contour lines are precise – so make sure this assumption is appropriate!

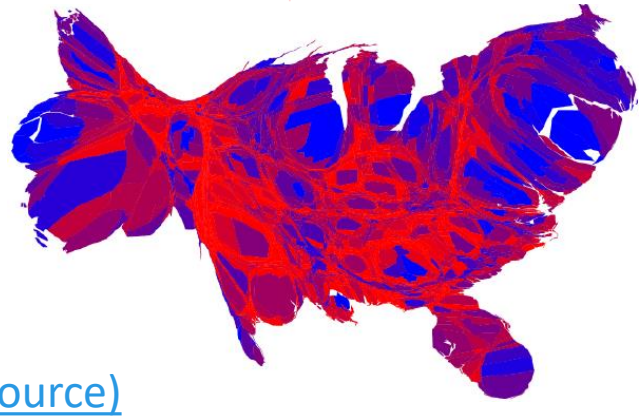


(source)

Gotcha #13:
Cartograms distort shapes



(source)



(source)

Cartograms (also: anamorphic maps) reshape the map areas:

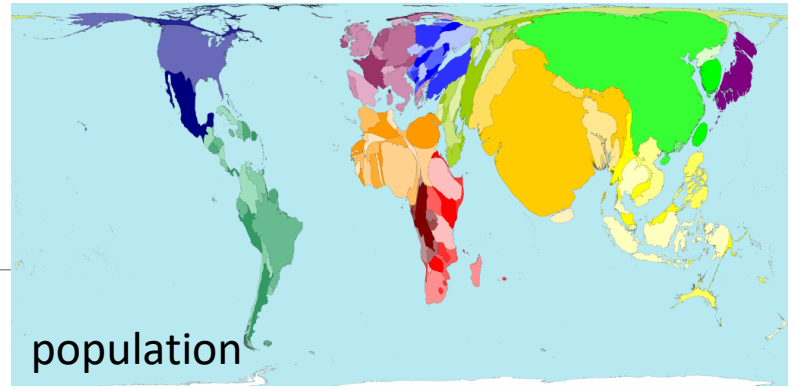
- Might use area to show hidden confounding variable (left)
- Might use area to show size of variable of interest (right)

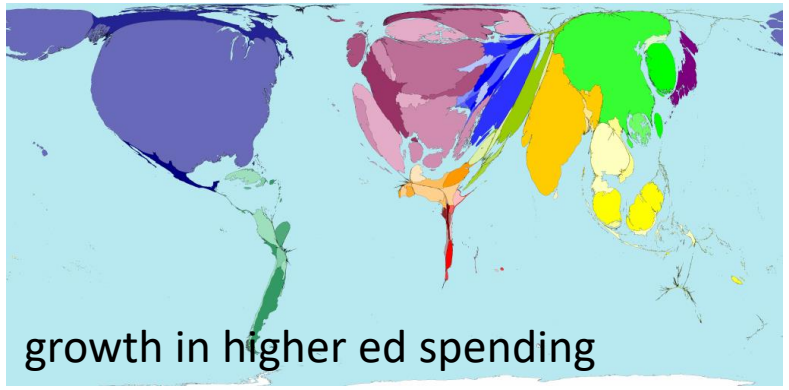
Cartograms address the human tendencies to associate area with importance

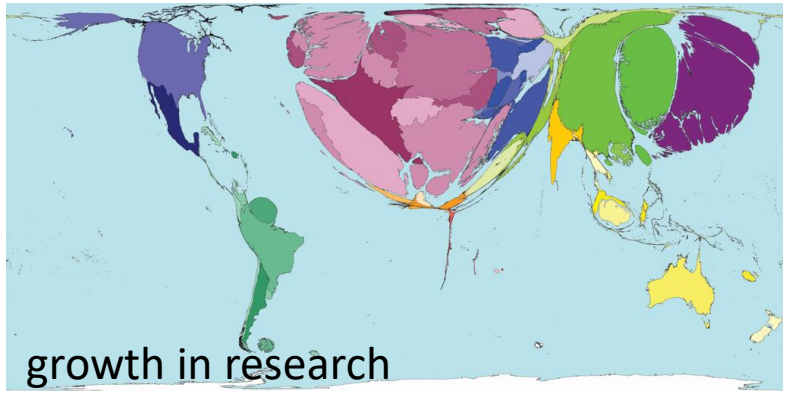
Cartograms rely on viewers' sense of geography – so consider using multiples and/or including a baseline!

http://www.worldmapper.org/

(source)







Gotcha #14:
All colors are not equal

Human perception matters

People can only distinguish 5-7 colors

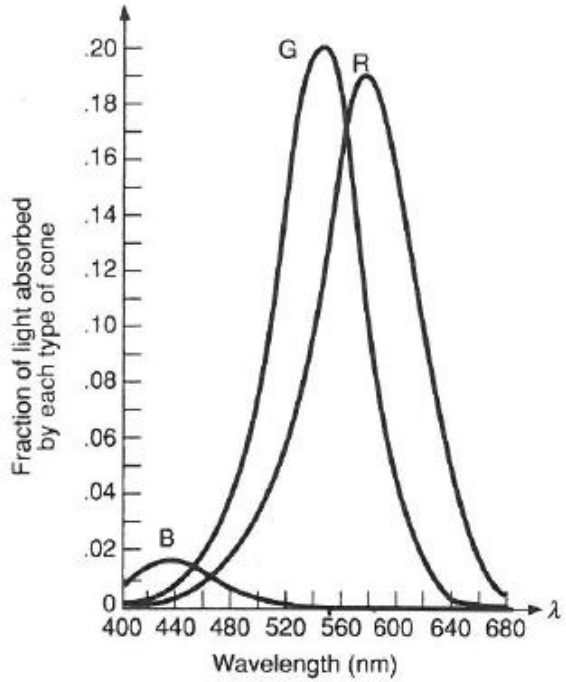
Perceptually uniform “colorspaces” (color schemes) won’t come from simplistic ramping (so: HSV or LAB >> RGB or CMYK)

Colorblindness simulators are available (example)
(and useful to good design & Section 508 Compliance)

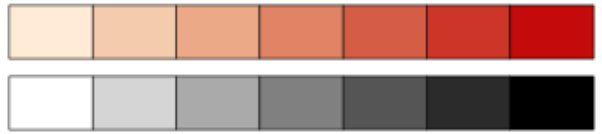
Convention matters

Discrete types should be in distinct colors

Continuous variables should be in hues of the same color



(source)

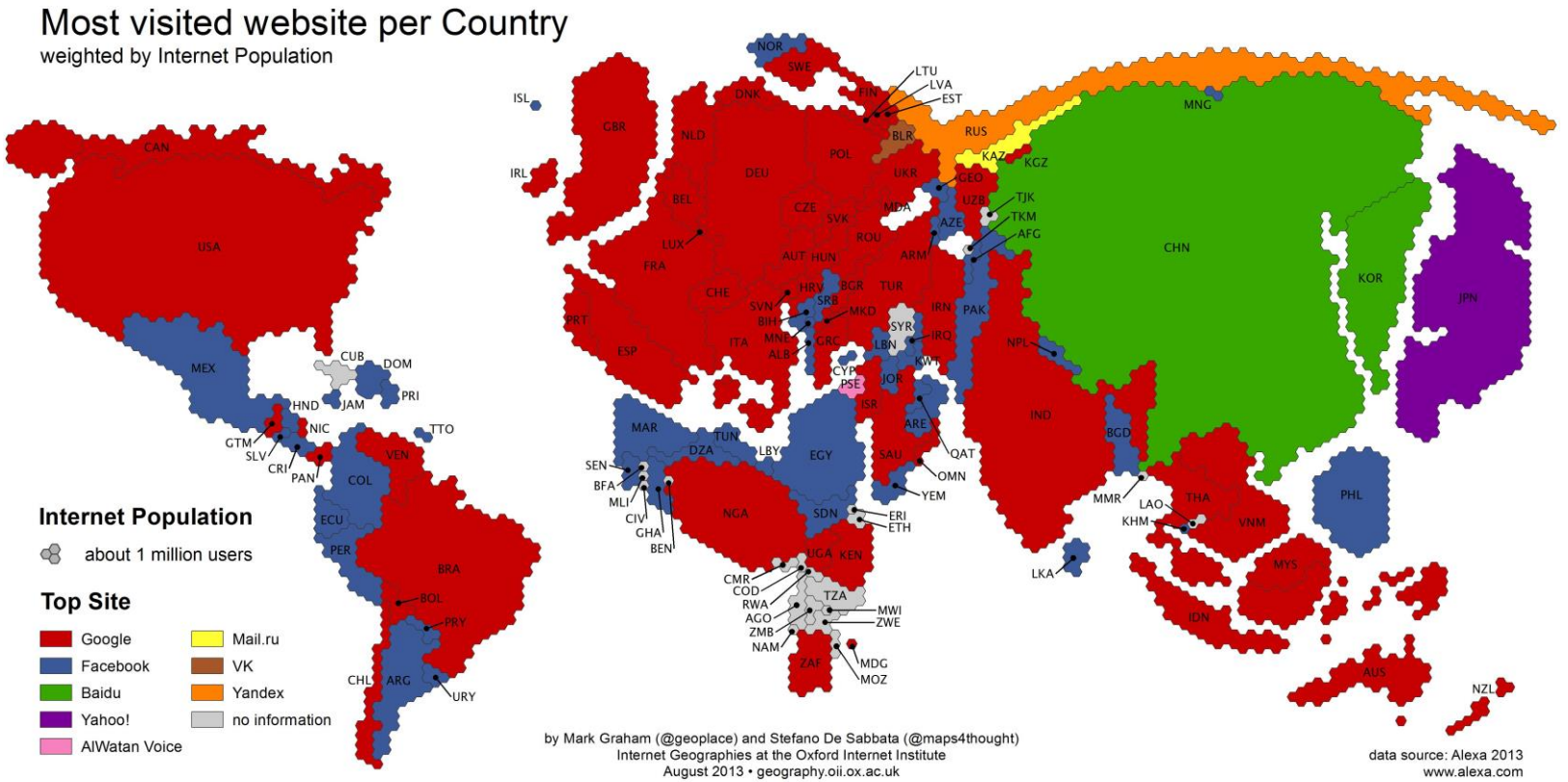


Magnitude (single hue progression)

Two extremes (bipolar hue progression)

Discrete data (full spectrum progression) (source)

Example: Map A

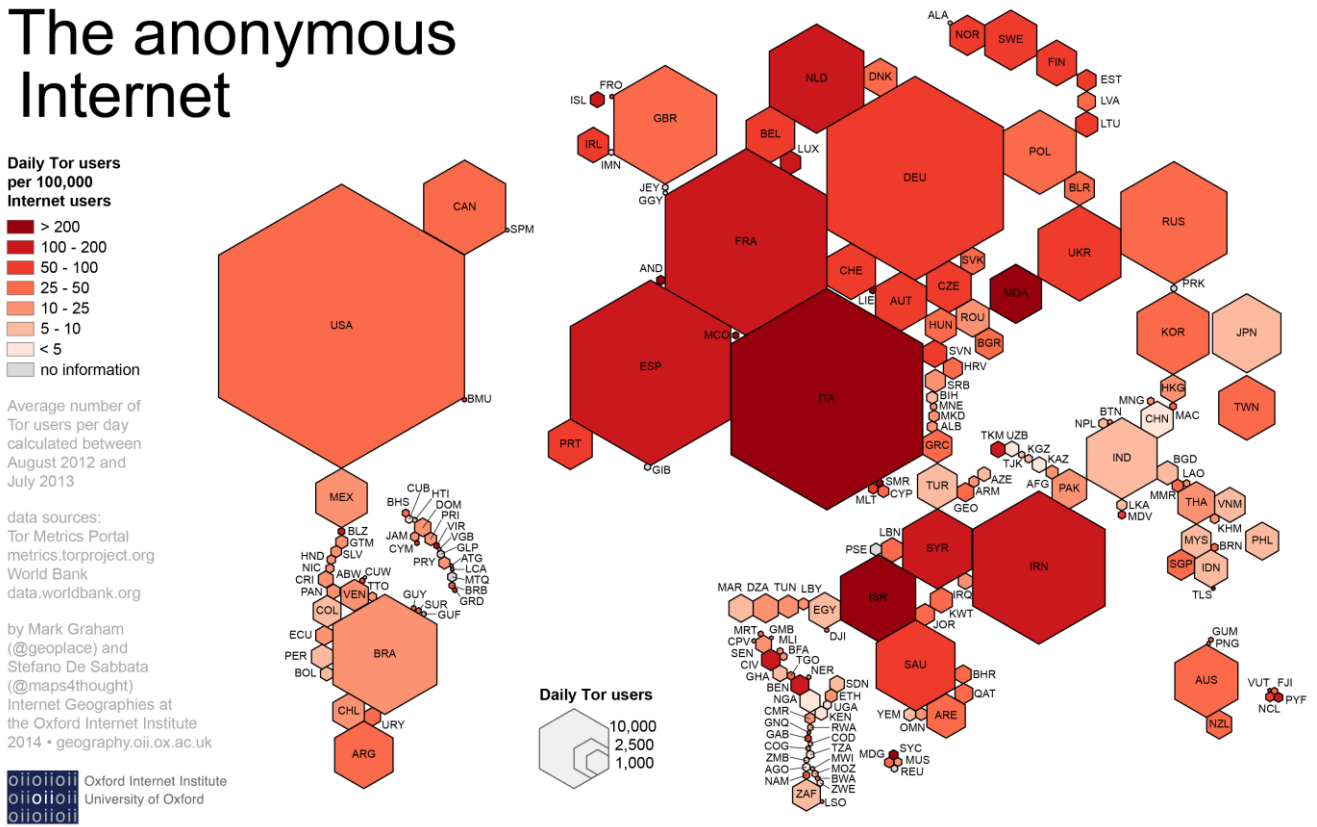


Can build maps that show multiple variables simultaneously (e.g., area, color, location)

Always be careful to normalize by confounding variables

(source)

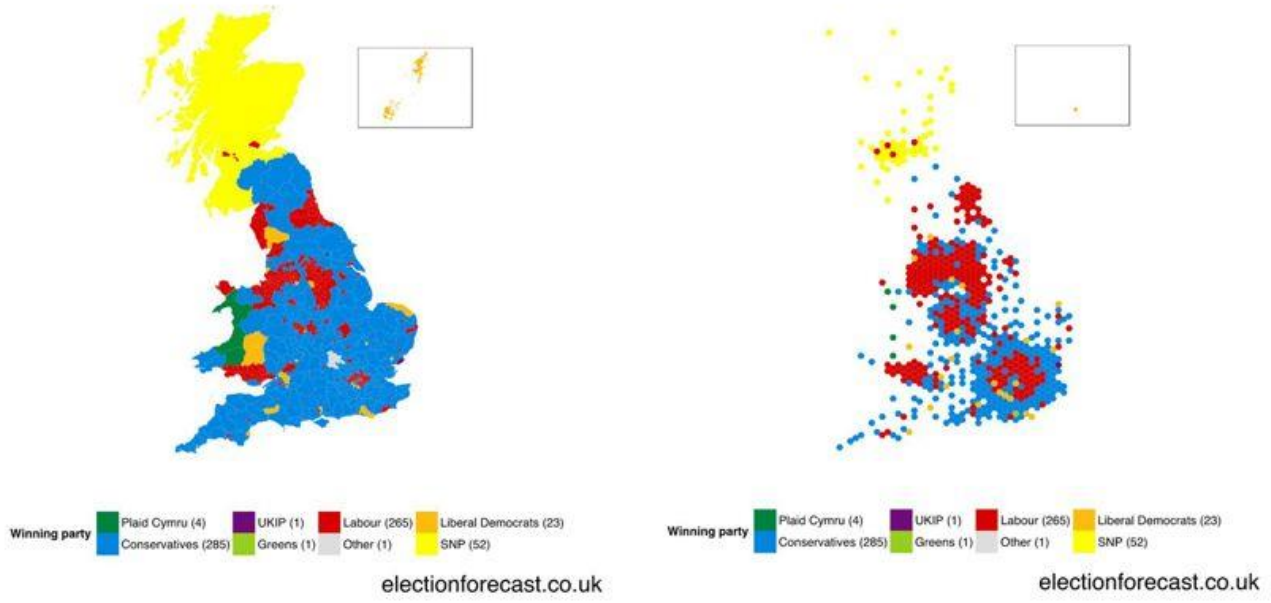
Example: Map B



Instead of deforming genuine boundaries, hexagons approximately placed might be simpler

[(source)]

Example: Map C



Instead of varying unit size, varying the number of a unit can reduce the implication of weight in empty space

(source)

Review of discussed content

Intro: How do we computationally represent geographic data?

Pro-tips / Gotchas:

- The best maps may still be flawed
- Use a spatial database
- Use hexagons to partition a surface
- Avoid the ecological fallacy & modifiable areal unit problem
- Do not analyze on lat-longs
- Divide out confounding variables
- Not all clusters are statistically significant
- Literal distance isn't always appropriate
- Spatial regression is tricky
- Consider network & other techniques
- Choropleths assume uniform potential
- Isopleths encourage assuming precision
- Cartograms distort shapes
- All colors are not equal

Conclusion: Let's reflect on sample maps that visualize spatial data