Introduction to maps with **R**

R Users Meeting Madrid · octubre2013

Beatriz Martinez

@_bmartinez_

What we need to plot a map?

What we need to plot a map?

Basically, plotting maps is:

What we need to plot a map?

Basically, plotting maps is:

- plot **points** in a given **coordinate system** (this points can be just points or become lines, polygons...)

What we need to plot a map?

Basically, plotting maps is:

- plot **points** in a given **coordinate system** (this points can be just points or become lines, polygons...)
- plot a **background map** to set a **reference** where those points are regarding to a region.

What we need to plot a map?

Basically, plotting maps is:

- plot **points** in a given **coordinate system** (this points can be just points or become lines, polygons...)
- plot a **background map** to set a **reference** where those points are regarding to a region.
- plot some **fact related** to every **point** (points, lines, polygons...)

What we need to plot a map?

Geographical data

'things' on the map

- points (e.g. parks and gardens)
- lines (e.g. metro lines)
- polygons (e.g. Madrid neighborhoods)

e.g.: Parks points

What we need to plot a map?

Geographical data

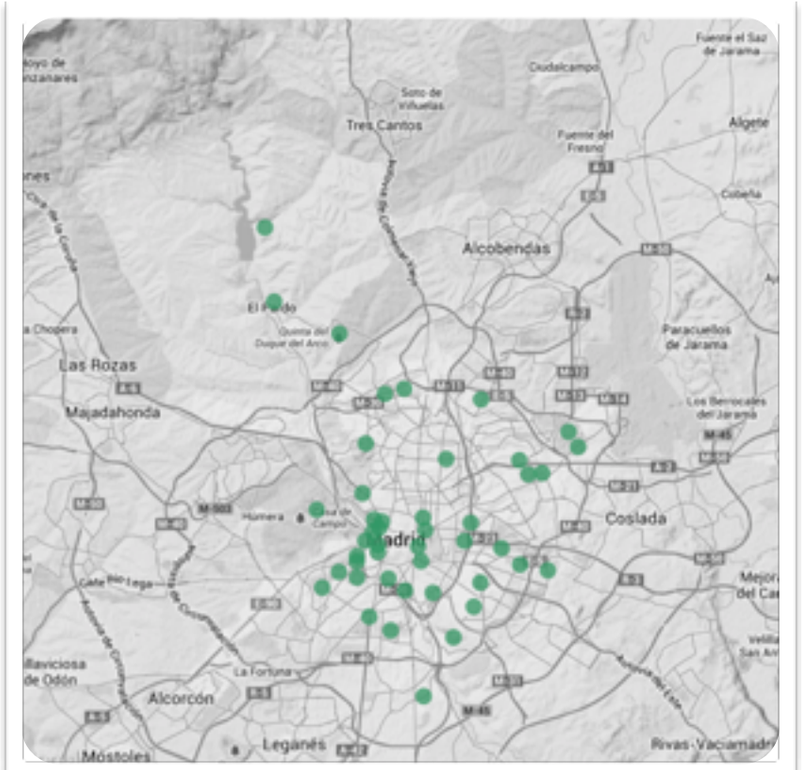
'things' on the map

- points (e.g. parks and gardens)
- lines (e.g. metro lines)
- polygons (e.g. Madrid neighborhoods)

Reference

to literally place the 'data'

- a background map
- a street map



e.g.: Parks points +

e.g.: Google Maps street map

What we need to plot a map?

Geographical data

'things' on the map

- points (e.g. parks and gardens)
- lines (e.g. metro lines)
- polygons (e.g. Madrid neighborhoods)

Reference

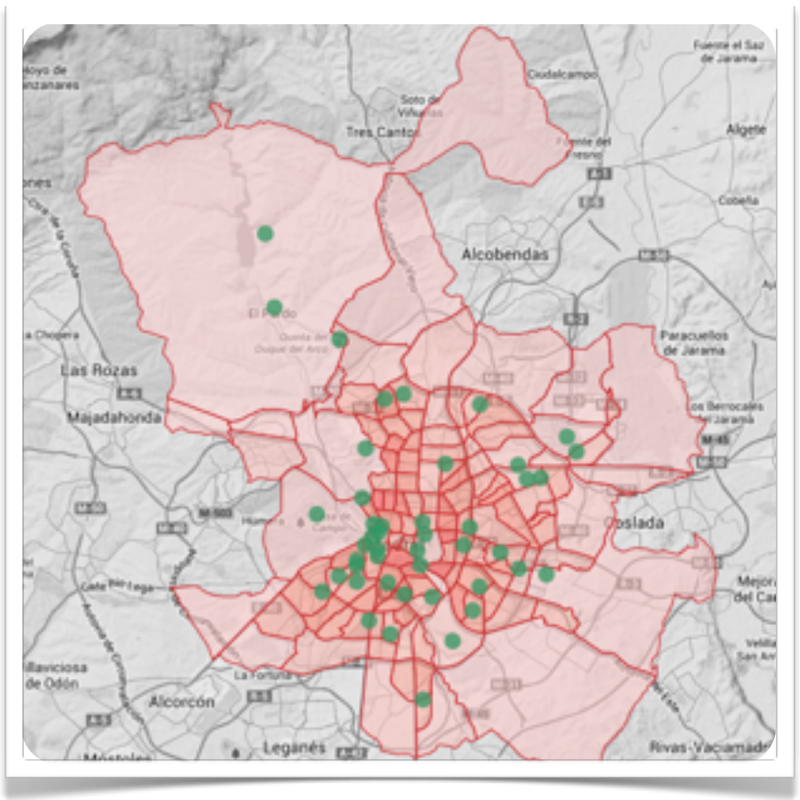
to literally place the 'data'

- a background map
- a street map

Meaningful data

attributes assigned to the geo data

- polygon (e.g.: population density)
- lines (e.g.: traffic flow in the routes)
- points (e.g. point of interest category)



e.g.: Parks points +

e.g.: Google Maps street map +

e.g.: Neighborhoods density

What we need to plot a map?

Geographical data

'things' on the map

- points (e.g. parks and gardens)
- lines (e.g. metro lines)
- **polygons (e.g. Spanish Autonomies)**

Reference

to literally place the 'data'

- a background map
- a street map

Meaningful data

attributes assigned to the geo data

- polygon (e.g.: population density)
- lines (e.g.: traffic flow in the routes)
- points (e.g. point of interest category)

e.g.: Spanish Autonomies +

e.g.: Autonomies density

What we need to plot a map?

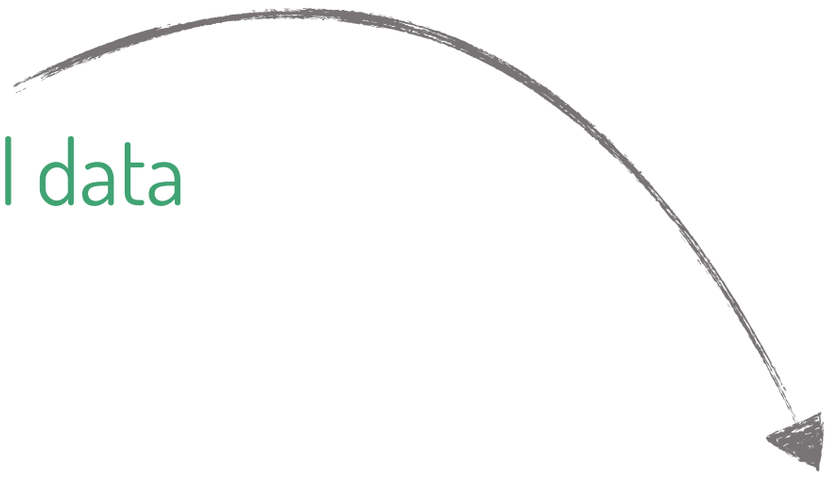
Geographical data

Reference

Meaningful data

What we need to plot a map?

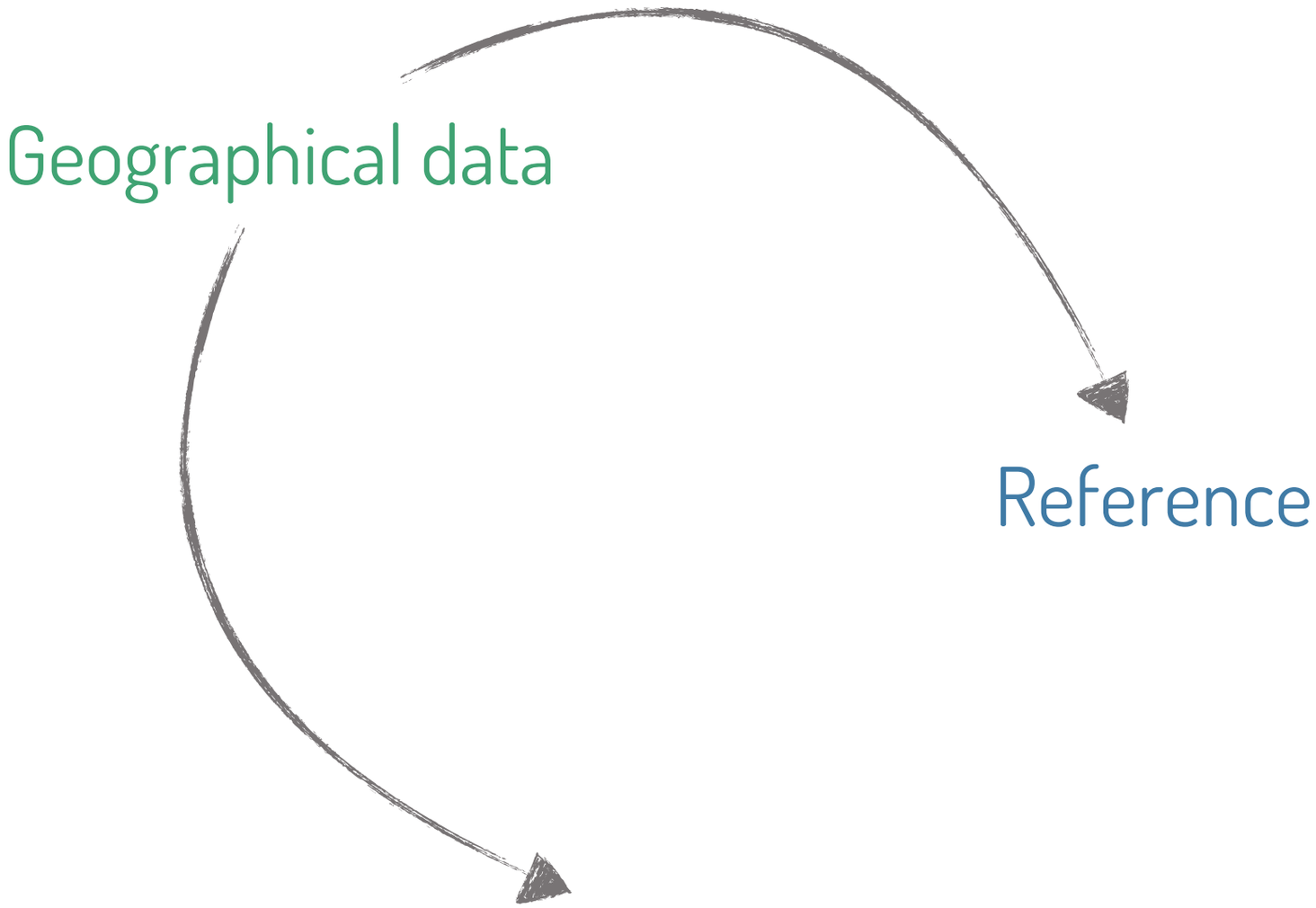
Geographical data



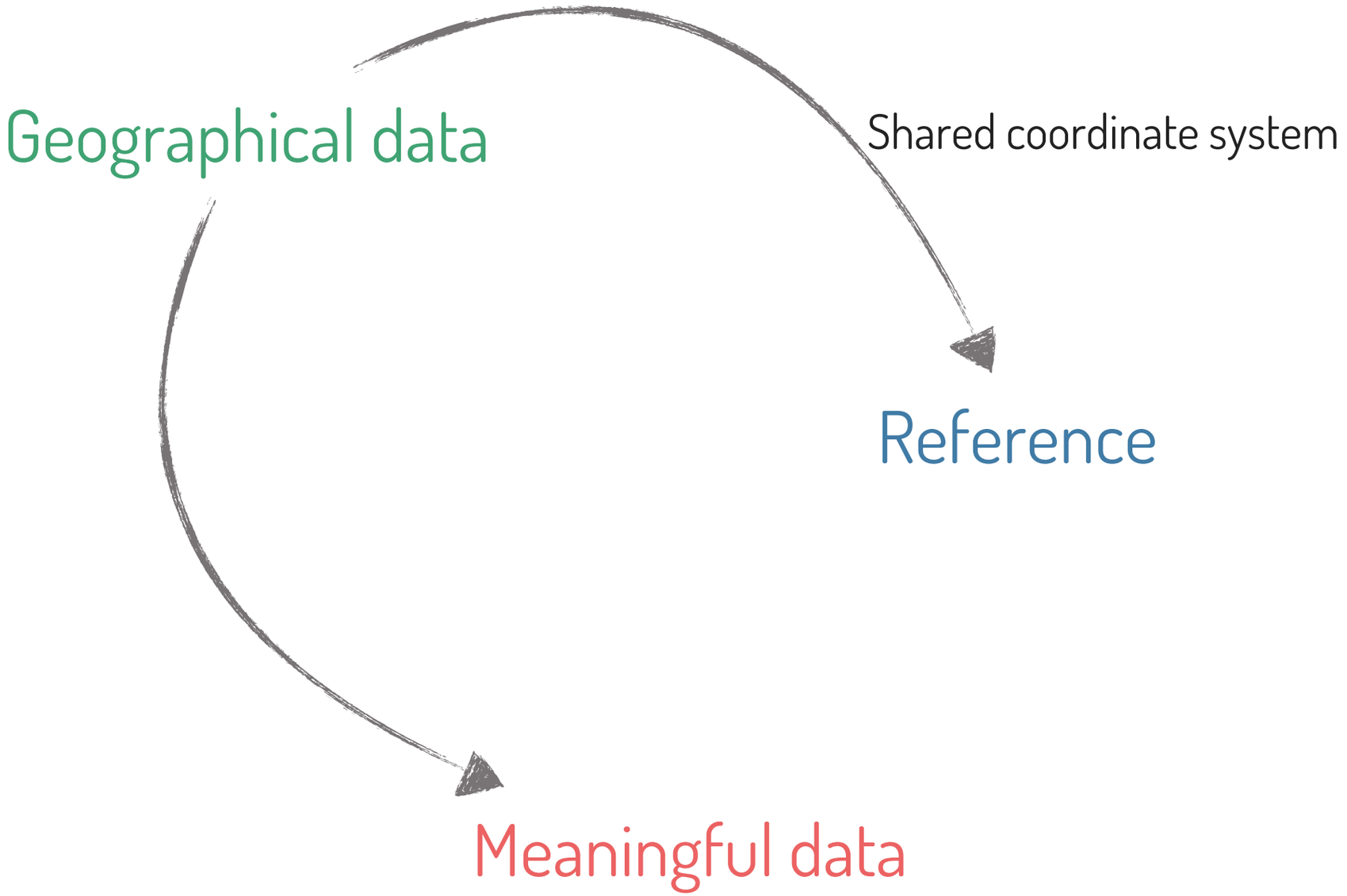
Reference

Meaningful data

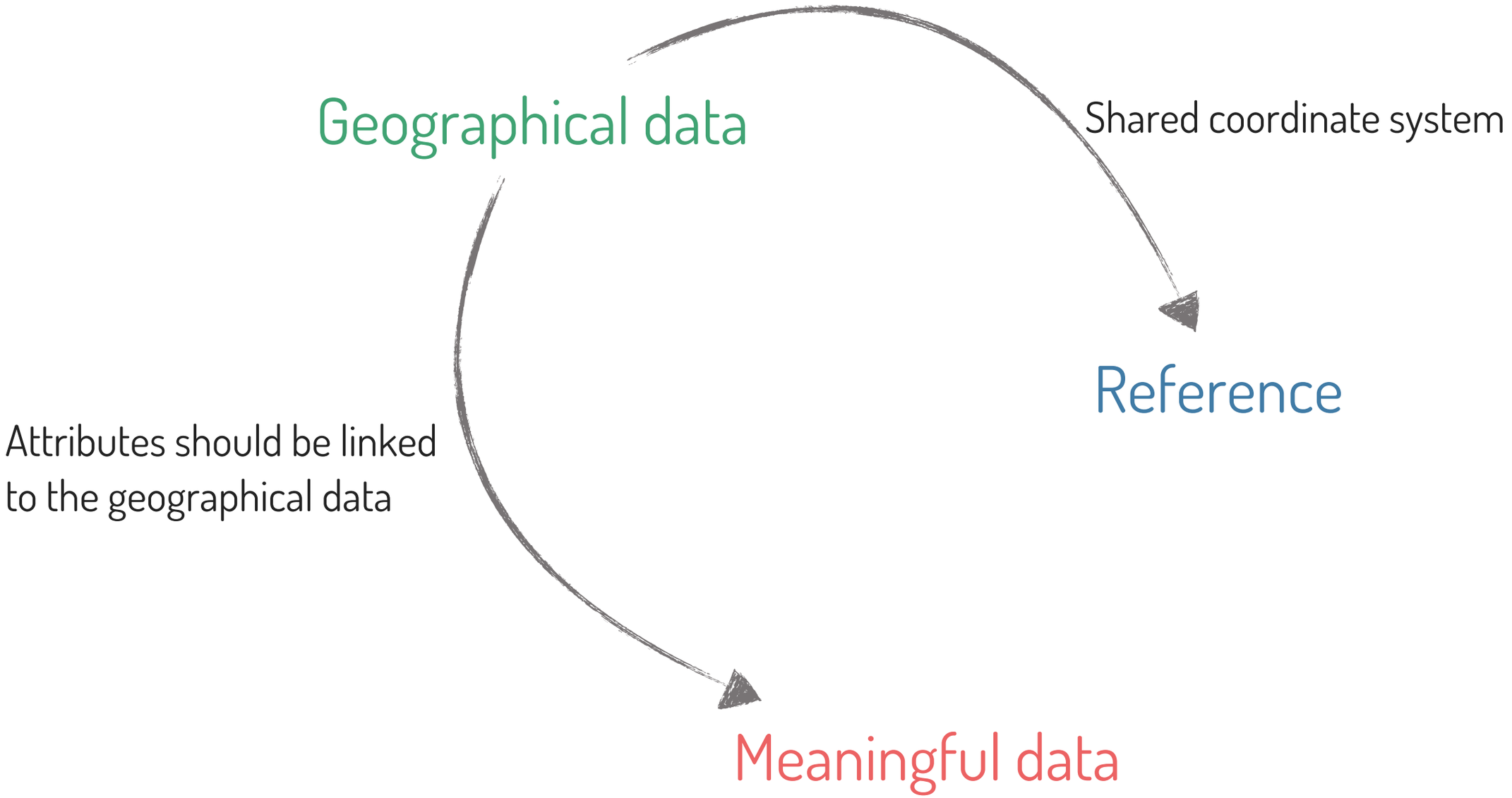
What we need to plot a map?



What we need to plot a map?



What we need to plot a map?



What we need to plot a map?

Geographical data

Shapefiles:

not just a table with the **coordinates** for the shape(points, lines, polygons...) but **also with attributes** (label for every polygon)

- **shp**: geometric objects
- **shx**: geometric object index
- **dbf**: geometric object attributes database

Attributes should be linked to the geographical data

Shared coordinate system

Reference

Meaningful data

some **R** libraries for maps

Library	description	objects' class	main functions	documentation
maps	▸ For drawing geographical maps. ▸ It has some geographical databases (some maps) ▸ does not read shapefiles (the most common *.shp file)	map	`map()` Plots a map. Create 'map' objects	http://cran.r-project.org/web/packages/maps/maps.pdf
mapdata	▸ Extra Map Databases	map	`map()`* *from the maps library	http://cran.r-project.org/web/packages/mapdata/mapdata.pdf

some **R** libraries for maps

Library	description	objects' class	main functions	documentation
sp	▸ **Classes and methods for spatial data** (polygons, lines, points...): **plotting** data as maps, spatial selection, retrieving **coordinates**, **set** or **change coordinate system**, **create Shape class objects**... ▸ It **has not geographical data**, but the methods and classes for **handling data** provided by **other sources**.	Spatial**DataFrame	`CRS(projargs)` Class "CRS" of coordinate reference system arguments `coordinates()` sets spatial coordinates to create spatial data, or retrieves spatial coordinates `bbox()` retrieve bbox from spatial data `spplot()` Lattice (trellis) plot methods for spatial data with attributes. `point.in.polygon()` do point(s) fall in a given polygon?	http://cran.r-project.org/web/packages/sp/sp.pdf
maptools	▸ Tools for **reading** and **handling spatial objects**. (shp, gps, klm...) ▸ It has the **wrld_simpl database** (world country polygons)	Spatial**DataFrame	`getinfo.shape()` Get shapefile header information. `readShape**()` Read arc shape files into Spatial**DataFrame objects `readShapeSpatial()` Read shape files into Spatial*DataFrame objects `map2SpatialPolygons()` Convert map objects to sp classes `pointLabel()` Label placement for points to avoid overlaps `dotsInPolys()` Fills polygons with a given amount of dots (e.g.: plot populations density with points instead of with a choropleth) `elide()` The elide function translate and disguise coordinate placing in the real world.(reflect, flip, rotate, shift....)	http://cran.r-project.org/web/packages/maptools/maptools.pdf

some **R** libraries for maps

Library	description	objects' class	main functions	documentation
rgdal	▸ access to coordinates/projection/ transformation operations	spatial	`spTransform()` for coordinates and map projection transformation	http://cran.r-project.org/web/packages/rgdal/rgdal.pdf
ggplot2	▸ **ggplot**: plots data.frames as if they were spatial objects.	data.frame	`fortify()` converts a generic R object (also spatial) into a data frame useful for plotting with ggplot2	http://docs.ggplot2.org/current/
ggmap	▸ **ggmap**: spatial visualization with Google Maps and OpenStreetMap	data.frame	`qmap()` gets a street map (googlempas, osm, stamen...) `geocode()` give the long lat coordinates for a locatios or a point of interest using Google Maps. `revgeocode()` retrieves an address (administrative area, country, postal code...) for a geocode `mapdist()` Compute map distances using Google Maps `route()` gives a route between two locations using Google Maps	http://cran.r-project.org/web/packages/ggmap/ggmap.pdf
googleVis	▸ provides an interface between R and the Google ▸ Chart Tools API. ▸ It allows users to create web pages with interactive ▸ charts based on R data frames	data.frame gvis	`gvisGeoChart()` reads a data.frame and creates text output referring to the Google Visualisation API. gvis objects can be plotted with the base function `plot()`	
rgeos	required for some functions from other libraries (e.g.: fortify)			

Demo R maps

Check out the **IntroductionR.R** file at
https://github.com/beamartinez/fuertehack

Some resources

- Madrid shapefiles

  http://www.madrid.org/nomecalles/DescargaBDTCorte.icm

- Spain shapefiles

  http://centrodedescargas.cnig.es/CentroDescargas/catalogo.do;jsessionid=332F8AE63FF7C8523368F8D31AF35066?destino=catalogo

  http://servicios2.marm.es/sia/visualizacion/descargas/mapas.jsp

  http://www.ine.es/ss/Satellite?L=en_GB&c=Page&cid=1254735116596&p=1254735116596&pagename=ProductosYServicios%2FPYSLayout

- Other shapefiles

  http://geocommons.com/

  http://gadm.org/country