Projections and distances

Abhi Datta

Department of Biostatistics, Bloomberg School of Public Health, Johns Hopkins University, Baltimore, Maryland

abhidatta.com @dattascience

Review of last lecture

- Spatial linear regression model for univariate point-referenced spatial data
- Modeling unknown surfaces with Gaussian Processes
- Kriging: Predictions at new locations
- Model comparison: AIC, BIC, RMSPE, CP, CIW
- Analysis in R using the geoR package

Using GP models for data referenced by latitude-longitude

- Data collected on the surface of the Earth are often referenced by latitude and longitude
- The Earth's surface is approximately ellipsoidal/spherical. How to visualize the data on a map?
- GP model covariance matrix requires pairwise distances between the location. What distance to use with latitude-longitude?

Map projections

An example dataset:

Lon	Lat	PM25
-76.53	39.36	15.61
-76.51	39.30	9.96
-76.59	39.30	9.77
-76.51	39.30	10.70
-76.61	39.27	11.27
-76.61	39.31	12.96
-76.72	39.38	13.13
-76.47	39.31	12.55
-76.60	39.30	9.87
-76.73	39.32	12.81
-76.63	39.41	9.03
-76.60	39.30	10.59
-76.64	39.36	11.91
-76.60	39.38	7.85
-76.45	39.20	9.75
-76.66	39.24	15.86
-76.44	39.21	10.23
-76.59	39.22	12.53
-76.63	39.37	11.82
-76.68	39.26	11.98
-76.68	39.32	12.69
-76.61	39.28	11.35
-76.47	39.31	12.04

Low-cost sensor $PM_{2.5}$ data (in $\mu g m^{-3}$) in Baltimore on July 4, 2019.

Map projections

An example dataset:

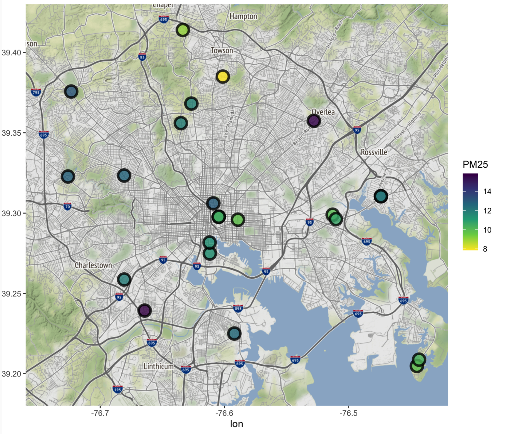


Figure: Low-cost sensor $PM_{2.5}$ data in Baltimore on July 4, 2019.

How were the long-lat data (angles) converted to points on a plane?

Map projections

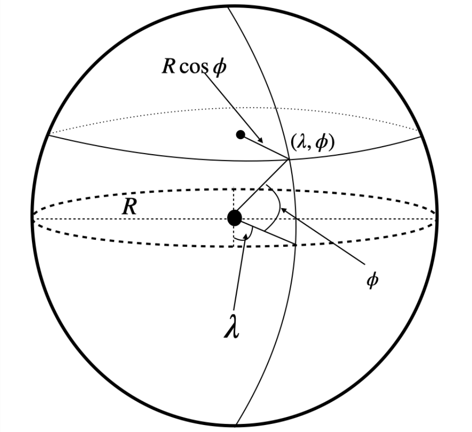
- How do we map 3-dimensional sphere $\mathbb{S}^2$ to a 2-dimensional plane $\mathbb{R}^2$?
- Basic idea: Find a 'reasonable' map (function) $h = (f, g) : \mathbb{S}^2 \to \mathbb{R}^2$ such that from longitude-latitude $(\lambda, \phi) \in \mathbb{S}^2$ we get the planar mapping $(x, y) = h(\lambda, \phi) \in \mathbb{R}^2$ with $x = f(\lambda, \phi)$, and $y = g(\lambda, \phi)$
- Can we find a distance-preserving map?

Map projections

- How do we map 3-dimensional sphere $\mathbb{S}^2$ to a 2-dimensional plane $\mathbb{R}^2$?
- Basic idea: Find a 'reasonable' map (function) $h = (f, g) : \mathbb{S}^2 \to \mathbb{R}^2$ such that from longitude-latitude $(\lambda, \phi) \in \mathbb{S}^2$ we get the planar mapping $(x, y) = h(\lambda, \phi) \in \mathbb{R}^2$ with $x = f(\lambda, \phi)$, and $y = g(\lambda, \phi)$
- Can we find a distance-preserving map?
  No! Gauss's Theorema Egregium[1] implies that a piece of paper cannot be bent onto a sphere without crumpling.
- All projections involve distortion of the distances

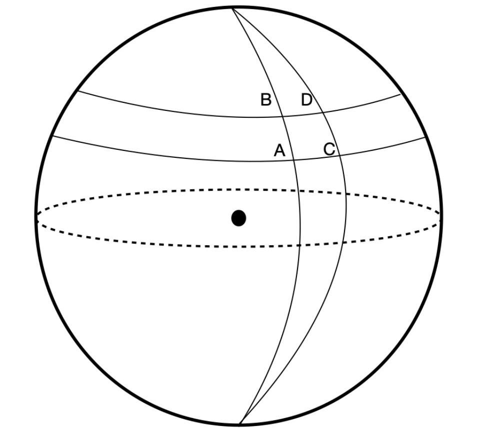
[1] https://en.wikipedia.org/wiki/Theorema_Egregium

Equal-area projections



Long-lat $= (\lambda, \phi)$, R=radius of the Earth

Equal-area projections



Consider an infinitesimally small patch ABCD,
A=(λ, ϕ), B=$(\lambda, \phi + d\phi)$, C=$(\lambda + d\lambda, \phi)$, D=$(\lambda + d\lambda, \phi + d\phi)$
with area $|AB||AC| = R^2 \cos\phi d\lambda d\phi$

Equal-area projections

With a planar map h, ABCD is mapped to A'B'C'D' where,
A'=$(f(\lambda,\phi), g(\lambda,\phi)$, B'=$(f(\lambda,\phi+d\phi), g(\lambda,\phi+d\phi))$,
C'=$(f(\lambda+d\lambda,\phi), g(\lambda+d\lambda,\phi))$,
D'=$(f(\lambda+d\lambda,\phi+d\phi), g(\lambda+d\lambda,\phi+d\phi))$.

Area of the parallelogram A'B'C'D' is the cross-product of
$\overrightarrow{A'C'} = (\partial f/\partial\lambda, \partial g/\partial\lambda)d\lambda$ and $\overrightarrow{A'B'} = (\partial f/\partial\phi, \partial g/\partial\phi)d\phi$

Equal-area map:
Area(A'B'C'D')=$(\partial f/\partial\lambda\ \partial g/\partial\phi - \partial f/\partial\phi\ \partial g/\partial\lambda)d\lambda d\phi =$
Area(ABCD)=$R^2 d\lambda d\phi$
$\implies \partial f/\partial\lambda\ \partial g/\partial\phi - \partial f/\partial\phi\ \partial g/\partial\lambda = R^2\cos\phi$

Equal-area projections

Example: sinusoidal projection:

$f(\lambda, \phi) = R\lambda \cos\phi, g(\lambda, \phi) = R\phi.$

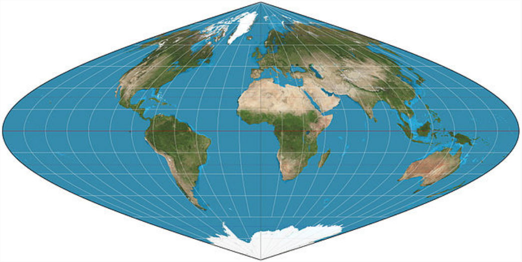


Figure: Sinusoidal projection[2]

The projection has equally spaced parallels

[2] https://en.wikipedia.org/wiki/Sinusoidal_projection

Conformal projections

Angle between AB and AC $= \pi/2$

Cosine of angle between A'B' and A'C' is proportional to the dot-product of $\overrightarrow{A'B'} = (\partial f/\partial\phi, \partial g/\partial\phi)d\phi$ and $\overrightarrow{A'C'} = (\partial f/\partial\lambda, \partial g/\partial\lambda)d\lambda$,
i.e., $\cos(\overrightarrow{A'B'}, \overrightarrow{A'C'}) \propto (\partial f/\partial\lambda\, \partial f/\partial\phi + \partial g/\partial\lambda\, \partial g/\partial\phi)$

Conformal (angle-preserving) map:
$\cos(\overrightarrow{A'B'}, \overrightarrow{A'C'}) = \cos(\overrightarrow{AB}, \overrightarrow{AC}) = \cos(\pi/2) = 0$
$\implies \partial f/\partial\lambda\, \partial f/\partial\phi + \partial g/\partial\lambda\, \partial g/\partial\phi = 0$

Mercator projection

$f(\lambda, \phi) = R\lambda$, $g(\lambda, \phi) = R \log \tan(\pi/4 + \phi/2)$

Figure: Mercator projection [3]

- Popular conformal projection used in `ggmap`[4]
- Equally spaced meridians
- Distortion at the poles (not usable beyond $\pm 80^\circ$ latitudes)

[3] `https://en.wikipedia.org/wiki/Mercator_projection`

[4] Kahle, D., & Wickham, H. (2013). ggmap: Spatial Visualization with ggplot2. The R journal, 5(1), 144-161.

UTM projection

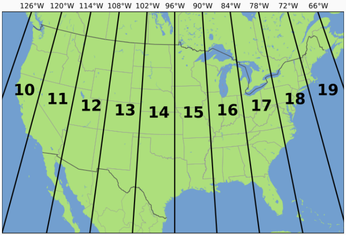


Figure: UTM zones for US[5]

- Universal transverse Mercator projection: Overlays a grid on the map – Divides the Earth into 60 zones each of 6-degree longitude, each zone is divided into 20 latitude bands
- Mercator projection for each zone
- Coordinates are given by easting and northing with respect to the grid

[5] https://en.wikipedia.org/wiki/Universal_Transverse_Mercator_coordinate_system

UTM projection

- long-lat to UTM conversion available in `spTransform` function of the `rgdal` R-package

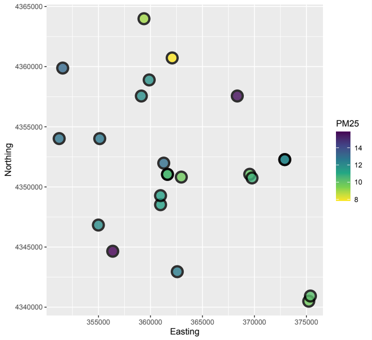
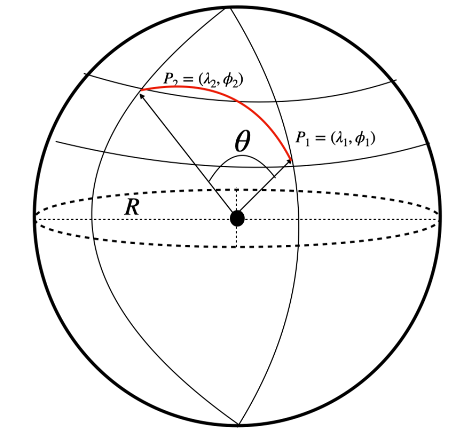


Figure: Low-cost sensor $PM_{2.5}$ data in Baltimore (UTM projections) on July 4, 2019.

Distances

- The GP model with isotropic covariance function uses pairwise distances between locations
- What distances to use when locations are on the Earth's surface?
- For many applications, variation of the outcome process along the surface of the Earth is of interest
- A natural candidate for distance is the geodesic distance or great circle distance – shortest distance between 2 points along the surface of the Earth

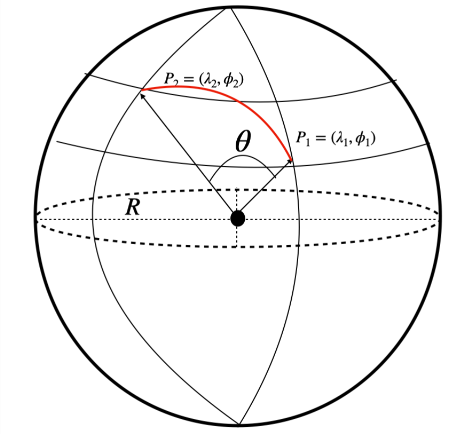
Geodesic Distance



Geodesic distance = great circle arc length = $R\theta$.

$cos(\theta) = \frac{<u_1, u_2>}{\|u_1\|\|u_2\|}$ where u_1, u_2 are the Euclidean co-ordinates corresponding to the polar co-ordinates (long-lat) P_1, P_2

Geodesic Distance



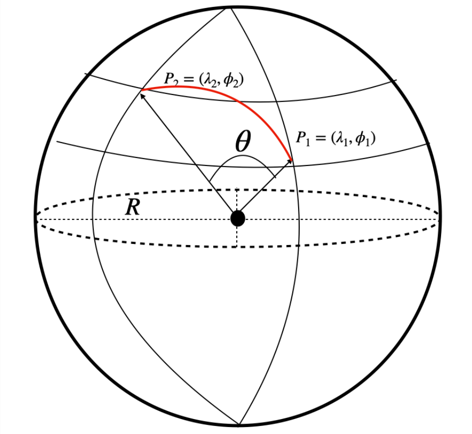
Geodesic distance = great circle arc length = $R\theta$.

$cos(\theta) = \frac{<u_1, u_2>}{\|u_1\|\|u_2\|}$ where

$u_i = (R\cos\lambda_i \cos\phi_i, R\sin\lambda_i \cos\phi_i, R\sin\phi_i) \implies$

$\cos\theta = \cos\phi_1 \cos\phi_2 \cos(\lambda_1 - \lambda_2) + \sin\phi_1 \sin\phi_2$

Geodesic Distance

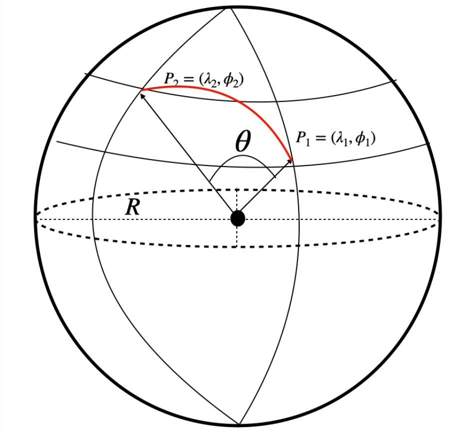


Geodesic distance = great circle arc length = $R\theta$.

$= R \arccos(\cos\phi_1 \cos\phi_2 \cos(\lambda_1 - \lambda_2) + \sin\phi_1 \sin\phi_2)$

`rdist.earth` function of the `fields` package calculates the geodesic distances given long-lat

Geodesic distances

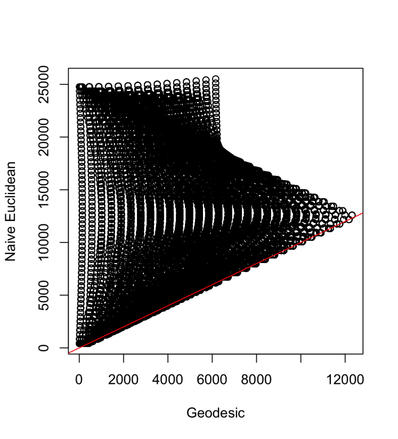


- Unfortunately, many popular covariance functions are not positive definite when using the geodesic distance on Earth
- E.g. Matérn covariance function with any smoothness $\nu > 1/2$ is not positive definite

Naive Euclidean distances

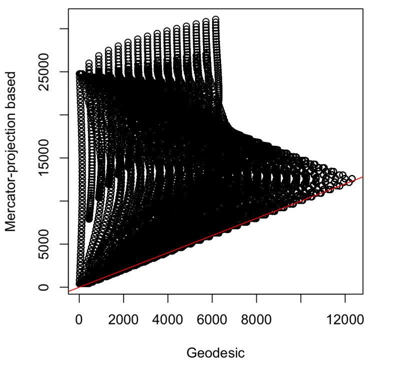
- Use directly the distance between the latitudes and longitudes
- $\|P_1 - P_2\| R\pi/180$ where $\|P_1 - P_2\| = \|(\lambda_1 - \lambda_2, \phi_1 - \phi_2)\|$
- Can be used with all covariances functions valid (positive-definite) on $\mathbb{R}^2$ (exponential, Matérn, Gaussian, etc.) as $(\lambda, \phi) \in (-180, 180] \times [-90, 90]$ is an Euclidean domain

Naive Euclidean distances



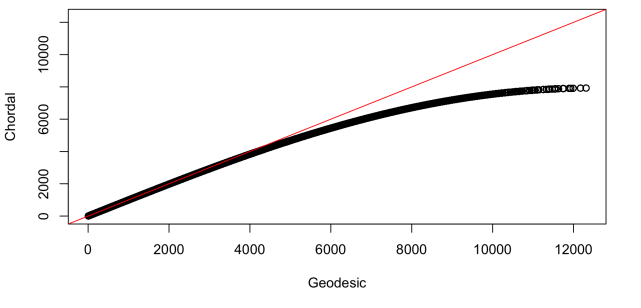
- Naive Euclidean distance always overestimates and greatly distorts the geodesic distance
- May be used only for small domains and away from the 180° meridian (infinite distortion)

Projection-based distances



- One can use the 2-dimensional Euclidean distance between point after using any projection
- For example, the Mercator projection may be used only for small domains and away from the poles (infinite distortion)
- The UTM Eastings and Northings can also be used for small domains

Chordal distance



- The Euclidean distance $\|u_1 - u_2\|$ between the points u_1 and u_2 gives the chord length between P_1 and P_2
- Any covariance function valid of $\mathbb{R}^3$ (exponential, Matérn, Gaussian, etc.) works with this chordal distance
- Will always underestimate the geodesic distance
- Moderate distortion even for large distances
- Maximum distortion $= \pi/2$ for two diametricaly opposite places

Summary

- Projections from the sphere $\mathbb{S}^2$ to the plane $\mathbb{R}^2$
- Equal-area maps, conformal maps, sinusoidal, Mercator and UTM projections, Eastings and Northings
- Plotting lat-long indexed data on maps
- Distances fo GP models – geodesic, naive Euclidean, projection-based, chordal
- When analyzing datasets covering large parts of the Earth, all these Euclidean distances lead to large distortions
- Large literature on constructing valid covariance functions directly on the surface of a sphere [6]

[6] Schoenberg, IJ (1942), 'Positive definite functions on spheres.', Duke Mathematical Journal, 9, 96–108
Jeong, J., & Jun, M. (2015). A class of Matérn-like covariance functions for smooth processes on a sphere. Spatial Statistics, 11, 1-18.