Volume 5, Issue 8, August 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Using Vincenty's Solutions of Geodesics on the Ellipsoid to Classify the Spatial Positioning of Individuals with Applications in Optimizing Data Collection of Residence

Varun Purohit
Delhi Technological University
India

Jigyasu Juneja
Fractal Analytics
India

Abstract— Geo-based analytics is becoming increasing popular with the enormous use of spatial and hyper-local mobile applications by individuals. Through this artefact we describe an algorithm for finding the most common geo spatial position from a given data set of positions with different accuracies. This paper particularly focuses on the implementation of algorithm using R as the statistical tool with IMAP as the package. The package helps in fast and accurate geo distance and spatial calculations using the Vincenty's solutions of geodesics on the ellipsoid. Finally we implement the algorithm on various data sets to understand the performance of the algorithm and discuss the results with potential application in accurately detecting the residence of an individual.

Keywords— Geo-distances, Spatial Positioning, Data Analysis, R, Algorithm, Geo-analytics

I. INTRODUCTION

This document is a template. An electronic copy can be downloaded from the Journal website. For questions on paper guidelines, please contact the journal publications committee as indicated on the journal website. Information about final paper submission is available from the conference website.

Volumes of Geo Data are growing at an exponential pace in the world with the introduction of geo based utility applications from supporting the rapidly increasing array of devices. This data is not being treated as a very important entity by any organisation for taking efficient business decisions with better ROIs.

With people using GPS technology and hyper-local services based on location becoming increasing common, geo-tracking for their benefits, location based analytics is becoming an important part for any business organisation. It is bridging the gap between the customer and services provider by providing the right service to the right customer at an appropriate time.

Geo Data can be captured in various forms through the devices a person carries. However, the most basic form of Geo Data is the latitude and longitude of a device on the per second basis. We use this information for finding the residence of the person as one of the applications of our algorithm.

A. Geo-Distances

There are various methods for calculating the geo distances between two points defined by latitude and longitude however no one is exactly accurate because of the various irregularities present on the earth surface. One of these methods though assume that surface curvature of the earth to be either flat, round or ellipsoidal. All these surface abstractions ignore the changes in elevation. The most accurate method which describes the distance between two points is the Vincenty solutions of geodesics on the ellipsoid which uses the ellipsoidal model of surface of the earth.

The reason for using the Vincenty's solution is because for the distance between two points on an ellipsoidal earth model is accurate to within 0.5mm distance on the ellipsoid being used. Calculations based on a spherical earth model, such as the haversine, are accurate to around 0.3%.

B. Overview of R

R is a language and environment for statistical computing and graphics. It is a GNU project which is similar to the S language and environment which was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues.

One of the biggest advantage of using R is that it is open source and is supported with various statistical packages such as for data manipulation and applying complex machine learning algorithms. These help in easy and fast handling and processing of data producing accurate and easy to visualize results.

C. Overview of IMAP package in R

IMAP is an interactive mapping package in R available on CRAN which uses various spatial locations in a map inspired for calculating Vincenty distance. Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations are used. The Geo-distance function, geodist() in this package can be used for calculating the distance between two positional coordinates with great accuracy and speed.

II. ALGORITHM DESIGN

Proposed algorithm comprises of two parts, simple frequency count and modified frequency count.

A. Simple Frequency Count

This is a simple frequency count algorithm. For n data points, n sized vector is used for the count. Each point can be imagined as a circle on a 2-D plane with its radius as the accuracy. Frequency vector is initialized with 0 as each n values. For each point, frequency of all those points are incremented, who all fall within its circle. This procedure can be summarised as, "To count the number of circles around a point".

If we were to compare for point n and k, frequency count increment by 1 for k if distance(n,k) <= accuracy(n) *distance(n,k) can be found by using library "Imap" which implements Vincenty's solution.

If point A falls within a circle drawn with accuracy of point B, then the frequency count of point A is incremented by "1". This simple procedure gives out a frequency vector. If the highest frequency is singular then that particular point is the clear winner. But this is rarely the case. Conflict arises when multiple highest frequency points are obtained. When this situation arises, we switch over to the modified frequency count.

B. Modified Frequency Count

This incorporates the accuracy of the incrementing point while modifying the frequency vector. As observed from above, incrementing factor is 1 in simple frequency count. In this procedure the increment is inversely proportional to accuracy's square.

Accuracy of a point can be associated with the radius it is making on the 2-D plane. As a point covers an area, probability shall be distributed on the area rather than getting distributed over the linear distance. For example, if a point has an accuracy of 10m, we can imagine it as a circle making a radius of 10m, forming an area directly proportional to 10*10. Now, suppose we have 10*10 water balloons accommodated in this area. We fill one of the balloons with air and replace it with any random water balloon. Probability of finding the balloon filled with air comes out to be 1 in 10*10. Take another point which has an accuracy of 20m. Again imagine it as a point making a circle of radius 20m. Area is directly propositional to 20*20 and therefore, it can accommodate 20*20 water balloons. We fill one of the balloons with air and replace it with any random water balloon. Probability of finding the balloon filled with air comes out to be 1 in 20*20.

We can observe from above that probability decreases with increase in the area. Since area is associated with the radius, which represents the accuracy, we can conclude that probability decreases with the square of accuracy. A point with a big accuracy value implies poor pinpointing capabilities. A point with infinite accuracy will end up covering all the points. Bigger the accuracy value, less useful it gets. Therefore, the incrementing factor is inversely proportional to the accuracy square.

In simple frequency count incrementing factor of 1 was taken but in the modified frequency count, the accuracy of incrementing circle is also taken into account. If point A comes into circle formed by point B, the accuracy of the incrementing circle, that is B will also be considered for incrementing factor. This is discussed below,

If we were to compare for point n and k, increment factor - p*(1/(accuracy[n]^2)) where p is any constant frequency count increment by increment factor for k if distance(n,k) <= accuracy(n)

*distance(n,k) can be found by using library "Imap" which implements Vincenty's solution.

In the example below, after simple frequency count we have two points B and E with the highest frequency count.

For Modified frequency count we took p = 1, for no particular reason. Observe that the accuracy of the incrementing circle will also be taken into account. For B, we have 3 + [(1/400) + (1/100) + (1/400)], that is, simple frequency count of B + Modified frequency count incrementation by point A, B and C. For point E we have 3 + [(1/100) + (1/100) + (1/100)], that is, simple frequency count of E + Modified frequency count incrementation by point D, E and F. Notice that incrementing factor for A is (1/400) and for B is (1/100) which holds inverse proportionality to the accuracy's square.

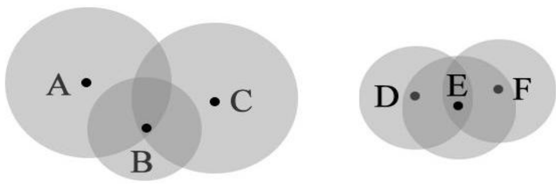

Fig. 1 Example for Simple and Modified frequency calculation

Table 1 Comparison of Simple and Modified Sample Frequency Count

Points	**A**	**B**	**C**	**D**	**E**	**F**
Accuracy	20	10	20	10	10	10
Simple Frequency Count	1	3	1	2	3	2
Modified Frequency Count, p=1	1.0025	3.0150	1.0025	2.0200	3.0300	2.0200

III. EXPERIMENT AND OBSERVATIONS

A. Algorithm - Residence

The aim of the algorithm is to find the most popular point in a given time frame.

B. Assumptions

Depending on the time frame selected, this can be the residence location during the night or work location during the day. One can assume that the person will be staying at home for the later part of night that is 12 midnight to 5:30 am.

C. Algorithm Pseudo-code

```
Residence Location <-  function (data for a device id) parameters:
        timelimit<- "a:b:c" #sampling range from 00:00:00 d<- Sys.Date()- n      #sampling days if(d< 1st available date in the dataset) d<- 1st available date in the dataset while(d<Sys.Date())
        {
        data for a day<- subset(data for a device id,date==d&time<timelimit)
                rows<- nrow(data for a day)
                if(rows==0)
                        next
                else
                        get the record of the best accuracy and bind it with latlong_new
                d<-d+1
        }
        sz<- nrow(latlong_new)
        if(sz==0)
        {
        Zero data
        break
        }
```

D. For all the points - simple frequency count

This is a simple frequency count of the number of circles in a particular latitude longitude point.

Frequency gets incremented by 1.

Circles are formed by the bounding circumference drawn with the accuracy associated with each latitude longitude point. if (number (max frequency) == 1) One clear Winner else.

E. For all the points - modified frequency count

This is a modified frequency count of the number of circles in a particular latitude longitude point. Frequency count increment take in account the accuracy of the circle which is causing the increment. Lower is the incrementing circle's accuracy more is the probability of the point's accuracy. Frequency gets incremented by (1/(incrementing circle's accuracy)^2) Circles are formed by the bounding circumference drawn with the accuracy associated with each latitude longitude point Point with the maximum frequency is the winner. Still if two points have same frequency, the one with the lower(better) accuracy is the winner.

F. Description

This can also be used to pinpoint the locations of interest or to map population distribution over an area.

The basic idea is to capture a time frame in which a person is found at home with the highest accuracy that is night. Time window starts from 12 midnight and is parametrised for the end time (05:30). All the points within this time window are captured and the point with the best accuracy out of all of them is picked and binded with latlong_new variable.

After the complete run all the daily best points are captured in latlong_new variable. Now, on all the points a simple frequency count is performed which calculates the number of circles in a point. If only one point comes out with the maximum frequency count, it is the winner. Else, a modified frequency count is performed which incorporates the incrementing circle's accuracy.

IV. RESULT

Test run was performed on database of over 3,00,000 points. This is a sample run for a particular device identified below by device-id. S.No$_1$ is the actual serial entry number in the database. S.No$_2$ is for explanation purpose.

As we are trying to capture the residence location, we'll sample one data point for each night. The number of points and the method of filtration of points can obviously be changed. We sampled one point per night within a defined time frame of midnight to 5:30 in the morning. Point with the best accuracy within this time frame is picked.

For a particular device id, these 42 data points were picked. Each data point has latitude, longitude, accuracy and timestamp as shown below. The initial part of the algorithm i.e. simple frequency count will be performed on these 42 points.

Table 2 Results for Simple Frequency Count

S.No$_1$	**S.No$_2$**	**latitude**	**longitude**	**accuracy**	**date**	**time**
18585	1	28.4522839	77.0599223	1060	18-12-2014	01:46:29
25703	2	28.4468626	77.08	1078	19-12-2014	00:04:35
36926	3	28.4468626	77.08	1078	20-12-2014	00:07:54
37092	4	28.4463407	77.0752677	1073	21-12-2014	00:43:57
45133	5	28.4434219	77.0740572	1111	23-12-2014	01:04:04
55301	6	28.4532656	77.0757838	1073	24-12-2014	00:08:11
66809	7	28.4283001	77.0588798	3.9	25-12-2014	00:00:00
77068	8	28.4449357	77.0782185	35	26-12-2014	00:07:18
82389	9	28.4475953	77.0790514	872	27-12-2014	00:06:23
94247	10	28.4476039	77.0790542	872	28-12-2014	00:00:05
105493	11	28.4476021	77.0790485	873	29-12-2014	00:04:16
125491	12	26.8599015	70.55806	2690	01-01-2015	00:03:35
133190	13	26.2715474	73.0097643	1505	02-01-2015	00:46:25
142626	14	28.4476208	77.0790425	876	03-01-2015	00:00:56
149095	15	28.5036253	77.0954156	882	04-01-2015	00:02:51
156986	16	28.4476596	77.0780118	1032	05-01-2015	00:04:19
162155	17	28.4472598	77.0791311	818	06-01-2015	00:04:44
169069	18	28.4472598	77.0791311	818	07-01-2015	00:03:55
171558	19	28.4472598	77.0791311	818	08-01-2015	00:12:43
174210	20	28.4437462	77.0738277	1152	09-01-2015	00:38:13
179033	21	28.4766475	77.0676665	983	10-01-2015	00:02:01
186561	22	28.4454267	77.0779805	25	12-01-2015	00:01:08
189757	23	28.4454325	77.0779506	23.41	13-01-2015	00:48:08
191170	24	28.4457127	77.077588	30	14-01-2015	00:06:22
194377	25	28.4453221	77.0780927	20	15-01-2015	00:46:55
194955	26	28.4435873	77.073747	1031	16-01-2015	02:05:56
198839	27	28.4464862	77.0801044	946	17-01-2015	02:23:47
201010	28	28.4462567	77.0798602	1028	18-01-2015	00:00:47
212571	29	28.4530278	77.0757825	975	19-01-2015	00:23:18
218909	30	28.4519121	77.0767867	1001	20-01-2015	00:38:09
222445	31	28.4463413	77.0798642	1037	21-01-2015	00:02:24
223960	32	28.4463334	77.0798544	1039	22-01-2015	00:04:43
231026	33	28.4478552	77.0775479	1041	23-01-2015	00:02:21
242429	34	28.4463252	77.0798564	1035	24-01-2015	02:22:13
245719	35	28.4470765	77.0800177	966	25-01-2015	03:27:39
247095	36	28.4457008	77.0777006	37.712	26-01-2015	00:01:01
249397	37	28.4436541	77.0738922	1019	27-01-2015	00:00:34
253023	38	28.4517296	77.0769007	988	28-01-2015	00:04:41
259649	39	28.4466406	77.0795404	1124	29-01-2015	00:04:29
265696	40	28.4517296	77.0769007	988	30-01-2015	02:42:44
280358	41	28.4962128	77.0881263	734	01-02-2015	00:01:20
289295	42	28.4466762	77.0795353	1123	02-02-2015	00:00:02

Points fed into algorithm : 42

After Simple frequency count we get the following array. Maximum frequency of 32 with it's 3 occurrences.

1_{1} 29_{2} 29_{3} 29_{4} 27_{5} 27_{6} 1_{7} 30_{8} 29_{9} 29_{10} 29_{11} 1_{12} 1_{13} 29_{14} 1_{15} 29_{16} 29_{17} 29_{18} 29_{19} 28_{20} 1_{21} 32_{22} 32_{23} 31_{24} 32_{25} 27_{26} 29_{27} 29_{28} 27_{29} 29_{30} 29_{31} 29_{32} 29_{33} 29_{34} 29_{35} 31_{36} 27_{37} 29_{38} 29_{39} 29_{40} 1_{41} 29_{42}

Count of maximum frequency - 3 [22,23,25]

Now, the competition is left between only these three points as they are of maximum frequency 32. Second part of the algorithm i.e. the modified frequency count will consider these three points.

Table 3 Results for Modified Frequency Count

$S.No_1$	$S.No_2$	latitude	longitude	accuracy	date	time
186561	22	28.4454267	77.0779805	25	12-01-2015	00:01:08
189757	23	28.4454325	77.0779506	23.41	13-01-2015	00:48:08
194377	25	28.4453221	77.0780927	20	15-01-2015	00:46:55

Modified Frequency Count
1.00000_1 29.00000_2 29.00000_3 29.00000_4 27.00000_5 27.00000_6 1.00000_7 30.00000_8 29.00000_9 29.00000_{10} 29.00000_{11} 1.00000_{12} 1.00000_{13} 29.00000_{14} 1.00000_{15} 29.00000_{16} 29.00000_{17} 29.00000_{18} 29.00000_{19} 28.00000_{20} 1.00000_{21} 32.00592_{22} 32.00592_{23} 31.00000_{24} 32.00592_{25} 27.00000_{26} 29.00000_{27} 29.00000_{28} 27.00000_{29} 29.00000_{30} 29.00000_{31} 29.00000_{32} 29.00000_{33} 29.00000_{34} 29.00000_{35} 31.00000_{36} 27.00000_{37} 29.00000_{38} 29.00000_{39} 29.00000_{40} 1.00000_{41} 29.00000_{42}

Modified frequencies for 22,23 and 25 are 32.00592, 32.00592 and 32.00592 respectively. As these are equal, the point with the best accuracy ie 25^{th} is declared as the residence point.

Maxplace : 25

Lat : 28.44532 Long : 77.07809

Accuracy: 20

V. CONCLUSION AND DISCUSSIONS

We have discussed the algorithm in detail and how in combination with the package IMAP solves a very big problem of efficiently tracking the position of an individual by finding most popular latitude-longitude point in a given data set. Although the example assumes that a person will be at home during the time period of 12:00 midnight to 05:30 morning, this assumption is bolstered by the time frame of the data set we have used. This algorithm can be further used to define the work place on an individual, the most visited restaurant and so on for efficient geo-based analytics. This can help in various marketing applications with location based offers, hyper-local applications where a person can find everything around him just based on his location. This can further individual customer centric analytics for various organisations.

VI. APPENDIX

Table 4 Notations used

Symbol	Meaning
A	length of semi-major axis of the ellipsoid (radius at equator);
F	flattening of the ellipsoid;
$b = (1 - f)\, a$	length of semi-minor axis of the ellipsoid (radius at the poles);
φ_1, φ_2	latitude of the points;
$U_1 = \arctan[(1 - f) \tan \varphi_1]$, $U_2 = \arctan[(1 - f) \tan \varphi_2]$	reduced latitude (latitude on the auxiliary sphere)
$L = L_2 - L_1$	difference in longitude of two points;
λ_1, λ_2	longitude of the points on the auxiliary sphere;
α_1, α_2	forward azimuths at the points;
A	azimuth at the equator;
S	ellipsoidal distance between the two points;
Σ	arc length between points on the auxiliary sphere

Inverse problem

Given the coordinates of the two points (φ_1, L_1) and (φ_2, L_2), the inverse problem finds the azimuths α_1, α_2 and the ellipsoidal distance s.

Calculate U_1, U_2 and L, and set initial value of $\lambda = L$. Then iteratively evaluate the following equations until λ converges:

$$\sin \sigma = \sqrt{(\cos U_2 \sin \lambda)^2 + (\cos U_1 \sin U_2 - \sin U_1 \cos U_2 \cos \lambda)^2}$$

$$\cos \sigma = \sin U_1 \sin U_2 + \cos U_1 \cos U_2 \cos \lambda$$

$$\sigma = \arctan (\sin \sigma / \cos \sigma)$$

$$\sin \alpha = (\cos U_1 \cos U_2 \sin \lambda) / \sin \sigma$$

$$\cos^2 \alpha = 1 - \sin^2 \alpha$$

$$\cos (2\sigma_m) = \cos \sigma - ((2 \sin U_1 \sin U_2) / \cos^2 \alpha)$$

$$C = (f/16)\cos^2\alpha\,[4 + f(4 - 3\cos^2\alpha)]$$

$$\lambda = L + (1-C) f \sin\alpha\,\{\alpha + C\sin\sigma\,[\cos(2\sigma_m) + C\cos\sigma\,(-1 + 2\cos^2(2\sigma_m))]\}$$

When λ has converged to the desired degree of accuracy (10^{-12} corresponds to approximately 0.06mm), evaluate the following:

$$u^2 = \cos^2\alpha\,((a^2 - b^2)/b^2)$$

$$A = 1 + (u^2/16384)\{4096 + u^2[-768 + u^2(320 - 175u^2)]\}$$

$$B = (u^2/1024)\{256 + u^2[-128 + u^2(74 - 47u^2)]\}$$

$$\Delta\sigma = B\sin\sigma\{\cos(2\sigma_m) + 0.25B[\cos\sigma(-1+2\cos^2(2\sigma_m)) - (1/6)B\cos(2\sigma_m)(-3+4\sin^2\sigma)(-3+4\cos^2(2\sigma_m))]\}$$

$$s = bA(\sigma - \Delta\sigma)$$

$$\alpha_1 = \arctan((\cos U_2 \sin\lambda)/(\cos U_1 \sin U_2 - \sin U_1 \cos U_2 \cos\lambda))$$

$$\alpha_2 = \arctan((\cos U_1 \sin\lambda)/(-\sin U_1 \cos U_2 - \cos U_1 \sin U_2 \cos\lambda))$$

Between two nearly antipodal points, the iterative formula may fail to converge; this will occur when the first guess at λ as computed by the equation above is greater than π in absolute value.

Direct Problem

Given an initial point (φ_1, L_1) and initial azimuth, α_1, and a distance, s, along the geodesic the problem is to find the end point (φ_2, L_2) and azimuth, α_2.

Start by calculating the following:

$$\tan U_1 = (1 - f)\tan\phi_1$$

$$\sigma_1 = \arctan(\tan U_1/\cos\alpha_1)$$

$$\sin\alpha = \cos U_1 \sin\alpha_1;\ \cos^2\alpha = (1 - \sin\alpha)(1 + \sin\alpha)$$

$$u^2 = \cos^2\alpha\,((a^2 - b^2)/b^2)$$

$$A = 1 + (u^2/16384)\{4096 + u^2[-768 + u^2(320 - 175u^2)]\}$$

$$B = (u^2/1024)\{256 + u^2[-128 + u^2(74 - 47u^2)]\}$$

Then, using an initial value $\sigma = s/bA$, iterate the following equations until there is no significant change in σ:

$$2\sigma_m = 2\sigma_1 + \sigma$$

$$\Delta\sigma = B\sin\sigma\{\cos(2\sigma_m) + 0.25B[\cos\sigma(-1+2\cos^2(2\sigma_m)) - (1/6)B\cos(2\sigma_m)(-3+4\sin^2\sigma)(-3+4\cos^2(2\sigma_m))]\}$$

$$s = bA(\sigma - \Delta\sigma)$$

Once σ is obtained to sufficient accuracy evaluate:

$$\phi_2 = \arctan\{[\sin U_1\cos\sigma - \cos U_1 \sin\sigma\cos\alpha_1]/[(1 - f)\sqrt{\sin^2 + (\sin U_1 \sin\sigma - \cos U_1\cos\sigma\cos\alpha_1)^2}]\}$$

$$\lambda = \arctan[(\sin\sigma\sin\alpha_1)/(\cos U_1\cos\sigma - \sin U_1\sin\sigma\cos\alpha_1)]$$

$$C = f/16\cos^2\alpha\,[4 + f(4 - 3\cos^2\alpha)]$$

$$L = \lambda - (1-C) f \sin\alpha\,\{\sigma + C\sin\sigma\,[\cos(2\sigma_m) + C\cos\sigma(-1 + 2\cos^2(2\sigma_m))]\}$$

$$\alpha_2 = \arctan(\sin\alpha/(-\sin U_1 \sin\alpha - \cos U_1\cos\alpha\cos\alpha_1))$$

If the initial point is at the North or South pole then the first equation is indeterminate. If the initial azimuth is due East or West then the second equation is indeterminate. If a double valued *atan2* type function is used then these values are usually handled correctly. Vincenty's modification [10]

In his letter to Survey Review in 1976, Vincenty suggested replacing his series expressions for A and B with simpler formulas using Helmert's expansion parameter k_1:

$$A = ((1 + 0.25(k_1)^2)/(1 - k_1))$$

$$B = k_1(1 - 3/8(k_1)^2)$$

where,

$$k_1 = [\sqrt{(1+u^2)} - 1]/[\sqrt{(1+u^2)} + 1]$$

Vincenty's paper presents a set of formulae rather than a single program; this shows those formulae pulled together as they are used in the script:

Where:

- s is the distance (in the same units as a & b)
- α_1 is the initial bearing, or forward azimuth
- α_2 is the final bearing (in direction $p_1 \rightarrow p_2$)

REFERENCES

[1] Aradhya Biswas, Goutham Pilla, and Bheemarjuna Reddy Tamma (2013) Microsegmenting, *An approach for precise distance calculation for GPS based ITS applications.*

[2] Ayman Al-Serafi,Ahmed Elragal (2013) *Trajectory Data Mining: a Novel Distance Measure*.

[3] Frank Ivis (2006) *Calculating Geographic Distance: Concepts and Methods*.

[4] Gennady Andrienko, Natalia Andrienko1, Daniel Keim, Alan M. MacEachren, and Stefan Wrobel (2011) *Challenging Problems of Geospatial Visual Analytics*.

[5] Giannotti, F., & Pedreschi, D., eds. (2007) *Mobility, Data Mining and Privacy - Geographic Knowledge Discovery. Springer, Berlin*.

[6] John R. Wallace (2012) *Package 'Imap' cran.r-project.org.*

[7] Stephen G Jones, Avery J Ashby, Soyal R Momin, and Allen Naidoo (2010) *Spatial Implications Associated with Using Euclidean Distance Measurements and Geographic Centroid Imputation in Health Care Research*.

[8] Stefan van der Spek, Jeroen van Schaick, Peter de Bois and Remco de Haan (2009) *Sensing Human Activity: GPS Tracking.*

[9] Thierry Mayer, Soledad Zignago (2005) *Notes on CEPII's distances measures: The GeoDist database.*

[10] T. Vincenty (1975) *Direct and Inverse solutions of geodesics on ellipsoid with applications of nested equations*.