

Distance on a globe

The shortest distance between two points is a straight line. This mathematical fact has even become a common expression in everyday speech. Technically, the distance in this term is the *Euclidian distance*. To calculate the distance between a point (x_1, y_1, z_1) and a point (x_2, y_2, z_2) in a three-dimensional space, you can use the following formula:

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}$$

However, on the round shaped earth, one has to confide in route definition and travel roads that follow the ground. Here, the shortest distance between two points remains a straight line. The only difference, is that this line is defined a bit different. On a globe, a straight line follows a big circle. Such a circle is the intersecting line of the globe with an area that intersects with the center of the globe.

There are different formulas to determine the shortest distance between two points on a globe. From a numerical point of view, the *haversine formula* is the most stable formula. The word *haversine* is a contraction of *halved versed sine*, where the *versed sine* of an angle θ equals $1 - \cos(\theta)$. If this is divided by two, that equals $\frac{1 - \cos(\theta)}{2} = \left(\sin\left(\frac{\theta}{2}\right)\right)^2$. If r represents the earth's radius, the *haversine formula* to determine the distance d between a point with co-ordinates (b_1, l_1) and a point with co-ordinates (b_2, l_2) is given by

$$d = 2r \arcsin\left(\sqrt{\frac{a}{1-a}}\right) \quad \text{where } a = \left(\sin\left(\frac{b_2 - b_1}{2}\right)\right)^2 + \cos(b_1)\cos(b_2)\left(\sin\left(\frac{l_2 - l_1}{2}\right)\right)^2$$

If you want to know the Euclidian distance between two points on the earth's surface, you first have to convert their co-ordinates to x , y and z co-ordinates. The formulas for this are very simple. If a point has the degree of latitude and the degree of longitude (b, l) , you can convert this to x , y and z co-ordinates with the following formulas

$$\begin{aligned} x &= r \cos(b) \cos(l) \\ y &= r \cos(b) \sin(l) \\ z &= r \sin(b) \end{aligned}$$

Here, r represents the earth's radius.

Assignment

- Write a function `latLng2Cartesian` that takes a degree of latitude and a degree of longitude in decimal degrees as an argument and that gives the x , y and z co-ordinate of that position in km. Caution: the geometric functions in the `math` module use angles and radians as an argument. However, there is a method that does a conversion to radians for you. Use 6,371 as an estimate for the earth's radius.

- Use the function `latLng2Cartesian` to write a function `euclideanDistance`, that takes two degrees of latitude and longitude in decimal degrees as an argument and returns the Euclidean distance between those points (in km)
- Write a function `haversineDistance` that takes two locations in decimal degrees of latitude and longitude as an argument and that returns the distance between two points on the globe's surface (in km). Use the *haversine formula* to calculate this distance.

Example

```
>>> latLng2Cartesian(0.0, 0.0)
(6371.0, 0.0, 0.0)
>>> euclideanDistance(0.0, 0.0, 0.0, 0.0)
0.0
>>> euclideanDistance(0.0, 0.0, 0.0, 180.0)
12742.0
>>> haversineDistance(0.0, 0.0, 0.0, 0.0)
0.0
>>> haversineDistance(0.0, 0.0, 0.0, 180.0)
20015.086796
```

De kortste afstand tussen twee punten is een rechte lijn. Dit wiskundig feit is zelfs een vaste uitdrukking geworden in de omgangstaal. Technisch gezien wordt de afstand waarvan hier sprake is de *Euclidische afstand* genoemd. Om in een driedimensionale ruimte de afstand d te berekenen tussen een punt (x_1, y_1, z_1) en een punt (x_2, y_2, z_2) , kan je de volgende formule gebruiken: $d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}$

Op de bolvormige Aarde moet men zich bij routebepaling echter beperken tot reiswegen die het aardoppervlak volgen. Daarbij blijft de kortste afstand tussen twee punten een rechte lijn. Alleen is een rechte lijn hier net iets anders gedefinieerd. Op een bol volgt een rechte lijn immers een grote cirkel. Zo'n grote cirkel is de snijlijn van de bol met een vlak dat door het middelpunt van de bol gaat.

Er bestaan verschillende formules om de kortste afstand tussen twee punten op een bol te bepalen. De *haversine formule* is echter vanuit numeriek oogpunt de meest stabiele. Het woord *haversine* is een samentrekking van *halfed versed sine*, waarbij de *versed sine* van de hoek θ gelijk is aan $1 - \cos(\theta)$. De helft hiervan is dan gelijk aan $\frac{1 - \cos(\theta)}{2}$ $= \left(\sin\left(\frac{\theta}{2}\right)\right)^2$. Als r de aardstraal voorstelt, dan wordt de *haversine formule* om de afstand d te bepalen tussen een punt met coördinaten (b_1, l_1) en een punt met coördinaten (b_2, l_2) gegeven door
$$a \quad \& \quad = \left(\sin\left(\frac{b_2 - b_1}{2}\right)\right)^2 + \cos(b_1)\cos(b_2)\left(\sin\left(\frac{l_2 - l_1}{2}\right)\right)^2 \\ c \quad \& \quad = \arctan\left(\sqrt{\frac{a}{1-a}}\right) \quad d \quad \& \quad = 2rc \quad \end{aligned}$$

Als je de Euclidische afstand tussen twee punten op het aardoppervlak wil weten, dan moet je eerst hun coördinaten omzetten naar x -, y - en z -coördinaten. De formules hiervoor zijn echter zeer eenvoudig. Als een punt breedte- en lengteligging (b,l) heeft, dan kan je dat omrekenen naar x -, y - en z -coördinaten met de volgende formules
$$\begin{aligned} x &= r \cos(b) \cos(l) \\ y &= r \cos(b) \sin(l) \\ z &= r \sin(b) \end{aligned}$$
 Hierbij stelt r de aardstraal voor.

Opgave

- Schrijf een functie `latLng2Cartesisch` die een breedte- en lengteligging in decimale graden als argument neemt en die de x -, y - en z -coördinaat van die positie in km teruggeeft. Let op: de goniometrische functies in de `math` module nemen hoeken in radialen als argument. Er is echter ook een methode beschikbaar die voor jou de omzetting naar radialen doet. Gebruik 6371 km als benadering voor de aardstraal.
- Gebruik de functie `latLng2Cartesisch` om een functie `euclidischeAfstand` te schrijven, die twee breedte- en lengteliggingen in decimale graden als argument neemt en de Euclidische afstand tussen die twee punten teruggeeft (in km).
- Schrijf een functie `haversineAfstand` die twee breedte- en lengteliggingen in decimale graden als argument neemt en de afstand teruggeeft tussen die twee punten over het boloppervlak (in km). Maak gebruik van de *haversine formule* om deze afstand te berekenen.

Voorbeeld

```
>>> latLng2Cartesisch(0.0, 0.0)
(6371.0, 0.0, 0.0)
>>> euclidischeAfstand(0.0, 0.0, 0.0, 0.0)
0.0
>>> euclidischeAfstand(0.0, 0.0, 0.0, 180.0)
12742.0
>>> haversineAfstand(0.0, 0.0, 0.0, 0.0)
0.0
>>> haversineAfstand(0.0, 0.0, 0.0, 180.0)
20015.086796
```