DistanceAndAzimuth

(Distance and direction between two points on Earth's surface)

Coded in *Mathematica* v. 7.0.0 by Vince Cronin
Revised December 5, 2018

Introduction

The purpose of this *Mathematica* notebook is to explain how to compute the direction or bearing from a reference point to another point, and to determine the shortest distance between those two points given the assumption that Earth is spherical.

Assumptions

We assume that Earth is a sphere of unit radius in all but one part of this analysis. Of course, Earth is not really a sphere. Earth has an irregular surface with deep ocean basins and tall mountains. The average radius of Earth is around 6,371.01±0.02 km (Yoder, 1995, p. 8). The deepest part of the ocean basins, called the Challenger Deep in the Mariana Trench, is ~11.0 km below sea level (that is, ~0.2% of Earth's mean radius), and the summit of Mt. Everest is ~8.848 km above sea level (~0.1% of Earth's mean radius). Earth's polar radius (6356.7530 km) differs from its mean equatorial radius (6378.1363 km) by only ~21 km, so Earth's shape is different from a sphere by about 1 part in 300 or about 0.3 % (*e.g.,* Cazenave, 1995, p. 36).

Assuming that Earth is a sphere of unit radius makes the mathematics we use in this problem much simpler than if we used either an ellipsoid or the geoid. For just one example, the radius of a sphere is the same measured at any point on the sphere's surface, and this is not true for either the reference ellipsoid or the geoid. Although Earth is a little wider at its equator than along its spin axis, we do not generate serious errors in our calculations by assuming Earth's radius is the same in all directions -- that it is a sphere.

When we say that Earth is assumed to have unit radius, that means that we set Earth's radius to 1 unit of length. The one part of this analysis in which we must use the actual size of the Earth in kilometers is in the determination of the circumferencial distance between the reference point and the observed pointt.

Specify the input data

Input only decimal degrees (*i.e.,* not degrees, minutes, seconds). The sign convention follows: north latitude is positive, south latitude is negative, east longitude is positive, west longitude is negative.

The variables **refPtLat** and **refPtLong** are the latitude and longitude of the reference point from which the distance and azimuth to the observed point is to be determined.

In[1]:=
```
refPtDescriptor = "NOAM pole";
```

In[2]:=
```
refPtLat = -4.85;
```

In[3]:=
```
refPtLong = -80.64;
```

The variables **otherPtLat** and **otherPtLong** are the latitude and longitude of the other point.

In[4]:=
```
otherPtDescriptor = "Nubia pole"
```

Out[4]=
```
Nubia pole
```

In[5]:=
```
otherPtLat = 47.68;
```

In[6]:=
```
otherPtLong = -68.44;
```

Operate on the input data

User-defined functions

The function **makeVector** computes the unit location vector of a given point on an assumed-spherical Earth, given that point's latitude and longitude. The x coordinate is equal to **Cos**[latitude]*__Cos__[longitude]. The y coordinate is equal to **Cos**[latitude]*__Sin__[longitude]. The z coordinate is equal to **Sin**[latitude]. The input latitude and longitude values must be in decimal degrees (*i.e.,* not degrees, minutes, seconds). The forms "*lat* **Degree**" and "*long* **Degree**" in the square brackets invokes the built-in *Mathematica* function **Degree** to convert the input data from degrees to radians, as required for the built-in *Mathematica* trigonometric functions (*e.g.*, **Sin**, **Cos**, **Tan**, **ArcSin**, **ArcCos**, **ArcTan**).

In[7]:=
```
makeVector[lat_, long_] := {Cos[lat Degree] Cos[long Degree],
   Cos[lat Degree] Sin[long Degree], Sin[lat Degree]};
```

The function **unitVector** finds the corresponding vector with a length of 1 (*i.e.,* unit length) given a vector of arbitrary length. It uses a built-in *Mathematica* function **Norm**, which returns the length of the input vector. Given a vector **A** with components $\{a_x, a_y, a_z\}$, the length or *norm* of **A** is equal to

$$\textbf{Norm}[\textbf{A}] = \sqrt{(a_x)^2 + (a_y)^2 + (a_z)^2}.$$

The unit-vector that corresponds to **A** has components $\{(a_x/\text{Norm}[\textbf{A}]), (a_y/\text{Norm}[\textbf{A}]), (a_z/\text{Norm}[\textbf{A}])\}$.

In[8]:=
```
unitVector[a_] := {a[[1]] / Norm[a], a[[2]] / Norm[a], a[[3]] / Norm[a]};
```

The built-in *Mathematica* function **VectorAngle** provides the angle between two vectors, in radians. Recall that 360° = $(2\,\pi)$ radians, so to convert from radians to degrees, multiply the radian value by $(180/\pi)$.

Computation

The circumference of a circle, whose radius (***r***) is known, is given by

$$\text{circumference} = 2 * \pi * \boldsymbol{r}$$

so given Earth's mean radius, in kilometers (Yoder, 1995)

In[9]:=
```
meanRadiusEarth = 6371;
```

Earth's circumference, in kilometers, is

In[10]:=
```
circumferenceEarth = 2 * π * meanRadiusEarth;
```

Define the unit location vector associated with Earth' s north pole (**northPole**)

In[11]:=
```
northPole = {0, 0, 1};
```

Utilize the user-defined fuction **makeVector** to convert the input latitude and longitude data into unit location vectors

In[12]:=
```
refPtVect = makeVector[refPtLat, refPtLong];
```

In[13]:=
```
otherPtVect = makeVector[otherPtLat, otherPtLong];
```

Find the distance from the reference point to the other point

The angular distance from the reference point to the other point, in radians, is ***θ1*** where

In[14]:=
```
θ1 = VectorAngle[refPtVect, otherPtVect];
```

The distance (expressed in kilometers) along the surface of an assumed-spherical Earth that corresponds to an angular distance of $\theta 1$ radians, is

In[15]:=
```
distanceKm = (θ1 / (2 * π)) * circumferenceEarth;
```

In[16]:=
```
distanceDeg = θ1 * (180 / π);
```

In[17]:=
```
distanceRad = θ1;
```

Find the azimuth from the reference point to the other point

Find the vector that is normal to the plane that contains the given two vectors, using a vector cross product. The built-in *Mathematica* function **Cross** performs this computation. If you had two vectors **A** and **B**, respectively specified by components {a_x, a_y, a_z} and {b_x, b_y, b_z}, the result of the vector cross product **A × B** would be a vector **C** with components {c_x, c_y, c_z} = {((a_y b_z)-(a_z b_y)), ((a_z b_x)-(a_x b_z)), ((a_x b_y)-(a_y b_x))} that is perpendicular to the the plane containing **A** and **B**. Given **A × B** = **C**, vector **B** is a right-handed rotation around **C** starting at **A**.

```
In[18]:= vectorAlpha = Cross[northPole, refPtVect];
```

```
In[19]:= vectorBeta = Cross[otherPtVect, refPtVect];
```

Utilize the built-in *Mathematica* function **VectorAngle** to find the angle ***θ*2** (expressed in degrees) between the two vectors **vectorAlpha** and **vectorBeta**, which is also the angular difference between [1] the azimuth from the reference point toward north, and [2] the azimuth from the reference point toward the other point.

```
In[20]:= θ2 = (180 / π) * (VectorAngle[vectorAlpha, vectorBeta]);
```

Find the angular distance (***θ*3**), in degrees, between the location vector to the other point (**otherPtVect**) and the **vectorAlpha**. This angle helps us define the bearing from the reference point to the other point.

```
In[21]:= θ3 = (180 / π) * (VectorAngle[vectorAlpha, otherPtVect]);
```

Imagine that Earth is divided into two hemispheres bounded by the plane that contains Earth's spin axis and the reference point. We will specify that the north pole is "up" in this frame of reference, so one hemisphere would be to the right of the reference point and the other would be to the left.
If the other point is located along the same circle of longitude as the reference point, then ***θ*3** will equal 90° or 180°, and the bearing from the reference point to the other point will either be 0° or 180°.
If the other point is located on the hemisphere to the right of the reference point, then ***θ*3** will be less than 90°. In that case, the azimuth or **bearing** from the reference point to the other point is equal to ***θ*2**. Bearings are expressed in degrees relative to north, where due north is 0°, due east is 90°, due south is 180°, and due west is 270°.
If the other point is located on the hemisphere to the left of the reference point, then ***θ*3** will be greater than 90°. In that case, the **bearing** from the reference point to the other point is equal to (360°- ***θ*2**).

The vector **vectorGamma** and the angle ***θ*4** are used when the other point is on the same meridian as the reference point, to differentiate between a bearing of 0° (north) or 180° (south) from the reference point to the other point. If ***θ*4** is less than 90°, the other point is north of the reference point. If ***θ*4** is equal to 90°, the other point is either coincident with the reference point or 180° from the reference point, in which case the bearing from the reference point to the other point is undefined. For simplicity, when ***θ*4**=90°, we set the bearing to 0°.

```
In[22]:= vectorGamma = Cross[refPtVect, vectorAlpha];
```

In[23]:=
```
θ4 = (180 / π) * (VectorAngle[vectorGamma, otherPtVect]);
```

The code fragment "If[((θ3 == 90) ∨ (θ3 == 180))" in the following line literally means "if θ3 equals 90 or if θ3 equals 180."

In[24]:=
```
bearing = If[((θ3 == 90) ∨ (θ3 == 180)),
   If[(θ4 <= 90), 0, 180], If[(θ3 > 90), (360 - θ2), θ2]];
```

Input Summary

Reference point {lat, long}

In[25]:=
```
refPtDescriptor
```

Out[25]= NOAM pole

In[26]:=
```
{refPtLat, refPtLong}
```

Out[26]= {-4.85, -80.64}

Other point {lat, long}

In[27]:=
```
otherPtDescriptor
```

Out[27]= Nubia pole

In[28]:=
```
{otherPtLat, otherPtLong}
```

Out[28]= {47.68, -68.44}

Output

The direction from the reference point to the other point (*i.e.,* the **bearing**) is given in degrees relative to north, where due north is 0°, due east is 90 °, due south is 180°, and due west is 270 °:

In[29]:=
```
bearing
```

Out[29]= 10.1794

The distance between the reference point and the other point is given in kilometers:

In[30]:=
```
distanceKm
```

Out[30]= 5961.82

The angular distance between the reference point and the other point is given in degrees:

In[31]:=
```
distanceDeg
```

Out[31]=
```
53.6159
```

The angular distance between the reference point and the other point is given in radians:

In[32]:=
```
distanceRad
```

Out[32]=
```
0.935774
```

References

Cazenave, A., 1995, Geoid, topography and distribution of landforms, *in* Ahrens, T.J., [editor], **Global Earth Physics -- A Handbook of Physical Constants**: American Geophysical Union Reference Shelf 1, p. 32-39, ISBN 0-87590-851-9.

Davis, H.F., and Snider, A.D., 1987, **Introduction to Vector Analysis** [5th edition]: Boston, Massachusets, Allyn and Bacon, 365 p., ISBN 0-205-10263-8.

Yoder, C.F., 1995, Astrometric and geodetic properties of Earth and the solar system, *in* Ahrens, T.J., [editor], **Global Earth Physics -- A Handbook of Physical Constants**: American Geophysical Union Reference Shelf 1, p. 1-31, ISBN 0-87590-851-9.