Package ‘wmm’

November 22, 2019

Type Package

Title World Magnetic Model

Version 1.0.0

Maintainer Will Frierson <will.frierson@gmail.com>

Description Calculate magnetic field at a given location and time according to the World Magnetic Model (WMM). Both the main field and secular variation components are returned. This functionality is useful for physicists and geophysicists who need orthogonal components from WMM. Currently, this package supports annualized time inputs between 2000 and 2020. If desired, users can specify which WMM version to use, e.g., the original WMM2015 release or the recent out-of-cycle WMM2015 release. Methods used to implement WMM, including the Gauss coefficients for each release, are described in the following publications: Chulliat et al (2019) <doi:10.25921/xhr3-0t19>, Chulliat et al (2015) <doi:10.7289/V5TB14V7>, Maus et al (2010) <https://www.ngdc.noaa.gov/geomag/WMM/data/WMMReports/WMM2010_Report.pdf>, McLean et al (2004) <https://www.ngdc.noaa.gov/geomag/WMM/data/WMMReports/TRWMM_2005.pdf>, and Macmil-lian et al (2000) <https://www.ngdc.noaa.gov/geomag/WMM/data/WMMReports/wmm2000.pdf>.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Depends R (>= 2.10)

RoxygenNote 6.1.1

Suggests testthat (>= 2.0.1), data.table (>= 1.12.2)

URL https://github.com/wfrierson/wmm

BugReports https://github.com/wfrierson/wmm/issues

Language en-US

NeedsCompilation no

Author Will Frierson [aut, cre]

Repository CRAN

Date/Publication 2019-11-22 11:20:02 UTC

R topics documented:

`.CalculateGaussCoef` *Lookup Table for Gauss coefficients g & h*

Description

Find Gauss coefficient $g_{n,m}(t)$ consistent with the World Magnetic Model.

Usage

```
.CalculateGaussCoef(t, t0, wmmVersion = "derived")
```

Arguments

`t`	Annualized date time. E.g., 2015-02-01 = (2015 + 32/365) = 2015.088
`t0`	Annualized reference time associated with `t`
`wmmVersion`	String representing WMM version to use. Must be consistent with `time` and one of the following: 'derived', 'WMM2000', 'WMM2005', 'WMM2010', 'WMM2015', 'WMM2015v2'. Default 'derived' value will infer the latest WMM version consistent with `time`.

Value

vector of Gauss coefficients, $g_{n,m}(t)$ and $h_{n,m}(t)$

`.CalculateGeocentricFieldSum`

Calculate sum of geocentric field components

Description

Calculate sum of geocentric field components

Usage

```
.CalculateGeocentricFieldSum(legendreTable, gaussCoef, radius, lon, latGC,
  deltaLatitude)
```

Arguments

`legendreTable`	`data.table` modified by `.CalculateSchmidtLegendreDerivative`
`gaussCoef`	Gauss coefficients as calculated by `.CalculateGaussCoef`
`radius`	Radius of curvature of prime vertical at given geodetic latitude
`lon`	GPS longitude
`latGC`	GPS latitude, geocentric
`deltaLatitude`	(Geocentric Latitude - Geodetic Latitude) in decimal degrees

`.CalculateMagneticField`

Calculate Expected Magnetic Field from WMM2015

Description

Calculate the magnetic field for a given location and time using the fitted spherical harmonic model from the 2015 World Magnetic Model.

Usage

```
.CalculateMagneticField(lon, latGD, latGC, radius, time,
  wmmVersion = "derived")
```

Arguments

`lon`	GPS longitude
`latGD`	GPS latitude, geodetic
`latGC`	GPS latitude, geocentric
`radius`	Radius of curvature of prime vertical at given geodetic latitude
`time`	Annualized date time. E.g., 2015-02-01 = (2015 + 32/365) = 2015.088

`wmmVersion`	String representing WMM version to use. Must be consistent with `time` and one of the following: 'derived', 'WMM2000', 'WMM2005', 'WMM2010', 'WMM2015', 'WMM2015v2'. Default 'derived' value will infer the latest WMM version consistent with `time`.

Value

Expected magnetic field from WMM2015 expressed as a vector, $m^{WMM}_{\lambda_t,\varphi_t,h_t,t}$

`.CalculateRadiusCurvature` *Radius of curvature of prime vertical*

Description

Calculate radius of curvature of prime vertical at given geodetic latitude.

Usage

```
.CalculateRadiusCurvature(latitudeGD)
```

Arguments

`latitudeGD`	Geodetic latitude in decimal degrees

Value

Radius of curvature of prime vertical at given geodetic latitude

---

`.CalculateRecursiveLegendre` *Run recursion to compute associated Legendre functions*

Description

Use recursion relations to compute the associated Legendre function, $P_{n,m}(\mu)$. User supplies degree n and order m as well as associated Legendre functions with smaller degree and order indices for recursion. When $n \leq 2$, $P_{n,m}(\mu)$ is directly calculated with known functions (i.e., no recursion). When $n > 2$, the following recursion relations are used based on the order m:

$$P_{n>2,m\leq 1}(\mu) = \frac{(2n-1)\cdot\mu\cdot P_{n-1,m}(\mu) - (n-1+m)\cdot P_{n-2,m}(\mu)}{(n-m)}$$

$$P_{n>2,m>1}(\mu) = \frac{2\mu(m-1)\cdot P_{n,m-1}(\mu)}{\sqrt{1-\mu^2}} - (n+m-1)\cdot(n-m+2)\cdot P_{n,m-2}(\mu)$$

Usage

```
.CalculateRecursiveLegendre(n, m, mu, index, Pn_1 = NULL, Pn_2 = NULL,
  Pm_1 = NULL, Pm_2 = NULL)
```

Arguments

n	Degree of associated Legendre function to compute
m	Order of associated Legendre function to compute
mu	Argument of associated Legendre function to compute
index	Index from `.kLegendreIndices` associated with `n` and `m`
Pn_1	$P_{n-1,m}(\mu)$
Pn_2	$P_{n-2,m}(\mu)$
Pm_1	$P_{n,m-1}(\mu)$
Pm_2	$P_{n,m-2}(\mu)$

Value

$P_{n,m}(\mu)$, scalar

`.CheckVersionWMM` *Check if given time is consistent with available WMM versions*

Description

Check if given time is consistent with available WMM versions

Usage

```
.CheckVersionWMM(t, wmmVersion)
```

Arguments

t	Annualized date time. E.g., 2015-02-01 = (2015 + 32/365) = 2015.088
wmmVersion	String representing WMM version to use. Must be consistent with `time` and one of the following: 'derived', 'WMM2000', 'WMM2005', 'WMM2010', 'WMM2015', 'WMM2015v2'.

.ConvertGeocentricToGeodeticFieldComponents
Geocentric Coordinates to Geodetic Coordinates

Description

Convert Geocentric Coordinates to Geodetic Coordinates.

Usage

```
.ConvertGeocentricToGeodeticFieldComponents(xGeocentric, yGeocentric,
  zGeocentric, deltaLat)
```

Arguments

`xGeocentric`	X-coordinate in geocentric system
`yGeocentric`	Y-coordinate in geocentric system
`zGeocentric`	Z-coordinate in geocentric system
`deltaLat`	(Geocentric Latitude - Geodetic Latitude) in decimal degrees

Value

Vector of length 3 representing geodetic coordinates consistent with given geocentric data

.ConvertGeodeticToGeocentricGPS
Convert from Geodetic to Geocentric Coordinates

Description

Convert geodetic coordinates to geocentric coordinates

Usage

```
.ConvertGeodeticToGeocentricGPS(latitudeGD, height)
```

Arguments

`latitudeGD`	Geodetic latitude in decimal degrees
`height`	Height in meters above ellipsoid (not mean sea level)

Value

List with first element as geocentric latitude in decimal degrees and second element as geocentric radius

`.DeriveVersionInfo` *Derive WMM version based on given time*

Description

Derive WMM version based on given time

Usage

```
.DeriveVersionInfo(t)
```

Arguments

`t`	Annualized date time. E.g., 2015-02-01 = (2015 + 32/365) = 2015.088

Value

List of reference year and compatible WMM versions inferred from `time`.

`.RunLegendreProcedure` *Compute Associated Legendre Functions Given Sequence of (degree, order) Indices*

Description

Procedure that computes the associated Legendre function, $P_{n,m}(\mu)$, given a sequence of (degree, order) indices and function argument μ. This is computed via recursive relationships for Legendre functions.

Usage

```
.RunLegendreProcedure(mu)
```

Arguments

`mu`	Function argument to $P_{n,m}(\mu)$

GetMagneticFieldWMM *Calculate Expected Magnetic Field from WMM*

Description

Function that takes in geodetic GPS location and annualized time, and returns the expected magnetic field from WMM.

Usage

```
GetMagneticFieldWMM(lon, lat, height, time, wmmVersion = "derived")
```

Arguments

`lon`	GPS longitude
`lat`	GPS latitude, geodetic
`height`	GPS height in meters above ellipsoid
`time`	Annualized date time. E.g., 2015-02-01 = (2015 + 32/365) = 2015.088
`wmmVersion`	String representing WMM version to use. Must be consistent with `time` and one of the following: 'derived', 'WMM2000', 'WMM2005', 'WMM2010', 'WMM2015', 'WMM2015v2'. Default 'derived' value will infer the latest WMM version consistent with `time`.

Value

list of calculated main field and secular variation vector components in nT and nT/yr, resp.: `x`, `y`, `z`, `xDot`, `yDot`, `zDot`

Examples

```
GetMagneticFieldWMM(
   lon = 240,
   lat = -80,
   height = 1e5,
   time = 2017.5,
   wmmVersion = 'WMM2015'
)

## Expected output
# X = 5683.51754 95763 nT
# Y = 14808.84920 23104 nT
# Z = -50163.01336 54779 nT

## Calculated Output
# X = 5683.518 nT
# Y = 14808.85 nT
# Z = -50163.01 nT
```

wmm *wmm: R Implementation of World Magnetic Model*

Description

The wmm package calculates magnetic field at a given location and time according to the World Magnetic Model.

WMM functions

This package has 1 exported function, GetMagneticFieldWMM, which returns a list of calculated main field and secular variation vector components in nT and nT/yr, resp.: x, y, z, xDot, yDot, zDot.

Acknowledgments

Thanks to:

- The WMM team past, present, and future for making the Gauss coefficients public domain
- Alex Breeze for tech reviewing the original version of this code, years ago

Index