Vmstools Reference Card

by Niels T. Hintzen, IMARES, part of Wageningen UR, niels.hintzen@wur.nl
2010-08-31. See also: http://code.google.com/p/vmstools/

Data

`data(eflalo2)` load eflalo2 test dataset
`data(tacsat)` load the tacsat test dataset
`data(harbours)` load the harbour test dataset
`data(VMShf)` load the VMS high ping rate test dataset
`data(VMS)` load the VMS test dataset

Metièr definitions

`Classif()`
`Fonctions()`

Tacsat Behavior Analyses

`filterTacsat(tacsat)` filter out records that do not lay within a speed range and/or change of heading interval
`pointInHarbour(tacsat)` flags tacsat points that are positioned in a harbour

Link eflalo2 - tacsat

`merge.vms.to.logbook.at.the.ping.scale(eflalo2,tacsatplus,general,vesselid)` Merge eflalo2 and tacsat+ on tacsat ping level
`mergeEflalo2Tacsat(eflalo2,tacsat)` Merge eflalo2 and tacsat at trip level

Interpolate tacsat

`interpolateTacsat(tacsat,interval,margin,res,method,params,headingAdjustment)` interpolate tacsat data between pings *x* minutes apart using straight line or cubic Hermite spline interpolation
`calculateCI(longitudes,latitudes,interpolations,indexInterpolation,tacsat,grid,spatialDataFrame,singleInterpolation,indexTacsat,parameters)` calculate a confidence interval around the interpolation
`diffInter(interpolation,tacsatHighRes)` calculate difference between true high-resolution data and interpolated dataset
`distanceInterpolation(interpolation)` calculate length of interpolation
`distanceTacsat(tacsat,index)` calculate distance between gps coordinates of a complete VMS dataset

Plotting

`createGrid(xrange,yrange,resx,resy)` create spatial grid
`mapGrid()`
`vmsGridCreate()`

Converting

`bearing(lon1,lat1,lon2,lat2)` calculate bearing from tacsat longitude and latitude data
`degree2Km(lon,lat,degree)` convert degrees to kilometers, only in longitudinal direction
`distance(lon1,lat1,lon2,lat2)` calculate distance between two gps coordinates
`km2Degree(lon,lat,km)` convert kilometers to degrees, only in longitudinal direction
`lonLatRatio(lon,lat)` compute the ratio between distance in longitude and latitude
`ICESrectangle(tacsat)` calculate ICES rectangle from gps location