Package: SailoR (via r-universe)

November 11, 2026

Type Package

Title An Extension of the Taylor Diagram to Two-Dimensional Vector Data

Version 1.3

Date 2026-10-30

Author Jon Sáenz [aut, cph] (ORCID: <https://orcid.org/0000-0002-5920-7570>), Sheila Carreno-Madinabeitia [aut, cph] (ORCID: <https://orcid.org/0000-0003-4625-6178>), Santos J. González-Rojí [aut, cre, cph] (ORCID: <https://orcid.org/0000-0003-4737-0984>), Ganix Esnaola [ctb, cph] (ORCID: <https://orcid.org/0000-0001-9058-043X>), Gabriel Ibarra-Berastegi [ctb, cph] (ORCID: <https://orcid.org/0000-0001-8681-3755>), Alain Ulazia [ctb, cph] (ORCID: <https://orcid.org/0000-0002-4124-2853>)

Maintainer Santos J. González-Rojí <santosjose.gonzalez@ehu.eus>

Description A new diagram for the verification of vector variables (wind, current, etc) generated by multiple models against a set of observations is presented in this package. It has been designed as a generalization of the Taylor diagram to two dimensional quantities. It is based on the analysis of the two-dimensional structure of the mean squared error matrix between model and observations. The matrix is divided into the part corresponding to the relative rotation and the bias of the empirical orthogonal functions of the data. The full set of diagnostics produced by the analysis of the errors between model and observational vector datasets comprises the errors in the means, the analysis of the total variance of both datasets, the rotation matrix corresponding to the principal components in observation and model, the angle of rotation of model-derived empirical orthogonal functions respect to the ones from observations, the standard deviation of model and observations, the root mean squared error between both datasets and the squared two-dimensional correlation coefficient. See the output of function UVError() in this package.

Depends R (>= 3.5.0)

License GPL-3

Encoding UTF-8

NeedsCompilation no

Repository https://sjgonzalezroji.r-universe.dev

Date/Publication 2026-11-05 15:37:15 UTC

RemoteUrl https://github.com/cran/SailoR

RemoteRef HEAD

RemoteSha 0a8f2b24050c150a47ecc72275dae37874034030

Contents

SailoR-package 2
Current 4
Dragonera 5
Ensembles 7
EstacaDeBares 11
Reanalysis 13
SailoR.Indices 15
SailoR.Plot 16
SailoR.Table 19
Synthetic 20
UVError 22
WRF 23

Index **26**

`SailoR-package` *An Extension of the Taylor Diagram to Two-Dimensional Vector Data*

Description

A new diagram for the verification of vector variables (wind, current, etc) generated by multiple models against a set of observations is presented in this package. It has been designed as a generalization of the Taylor diagram to two dimensional quantities. It is based on the analysis of the two-dimensional structure of the mean squared error matrix between model and observations. The matrix is divided into the part corresponding to the relative rotation and the bias of the empirical orthogonal functions of the data. The full set of diagnostics produced by the analysis of the errors between model and observational vector datasets comprises the errors in the means, the analysis of the total variance of both datasets, the rotation matrix corresponding to the principal components in observation and model, the angle of rotation of model-derived empirical orthogonal functions respect to the ones from observations, the standard deviation of model and observations, the root mean squared error between both datasets and the squared two-dimensional correlation coefficient. See the output of function `UVError()` in this package.

Author(s)

Jon Sáenz, Sheila Carreno-Madinabeitia, Santos J. González-Rojí, Ganix Esnaola, Gabriel Ibarra-Berastegi and Alain Ulazia

Maintainer: Santos J. González-Rojí <santosjose.gonzalez@ehu.eus>

References

J. Sáenz, S. Carreno-Madinabeitia, G. Esnaola, S.J. González-Rojí, G. Ibarra-Berastegi and A. Ulazia (2020). The Sailor diagram - A new diagram for the verification of two-dimensional vector data from multiple models. Geoscientific Model Development, 13(7), 3221-3240. doi:10.5194/gmd1332212020.

See Also

UVError

Examples

```
#-------------------------------------------------------
# Example 1: Figure 5 (left) from Sáenz et al. (2020)
#-------------------------------------------------------

# Load the data
data(Dragonera)

# Parameters
Uxlim=c(-10,10)
Uylim=c(-10,10)
Uxlab<-"U (m/s)"
Uylab<-"V (m/s)"
plotmain<-"Dragonera U10/V10"
sfactor<-1
ref<-Dragonera[["ref"]]
mod<-Dragonera[["mod"]]

# Index
SailoR.Indices(ref,mod)

# Plot
p <- SailoR.Plot(ref,mod,ColourList=NULL,sfactor,docenter=TRUE,
                 Uxlim,Uylim,Uxlab,Uylab,plotmain,plotlegend=TRUE,
                 plotRMSElegend=TRUE,
                 plotscalelegend=TRUE,
                 RMSE_legend_Rounding=1,RMSE_legend_units = " m/s",
                 referenceName="Reference")

#Add segments
segments(-1,-1, x1 = -1, y1 = 1,col="brown",lwd=4)
segments(-1,-1, x1 = 1, y1 = -1,col="brown",lwd=4)
segments(1,-1, x1 = 1, y1 = 1,col="brown",lwd=4)
segments(1,1, x1 = -1, y1 = 1,col="brown",lwd=4)
```

`Current` *Ocean Current Data*

Description

Contains the information of the surface ocean currents measured in the Bay of Biscay by the Donostia buoy and the HF-Radar from the Basque Meteorological Agency EUSKALMET. Additionally, it also contains model data from the high resolution global analysis and forecasting system PSY4V3R1 of the Copernicus Marine Environment Monitoring Service (CMEMS). Only the closest spatial grid points to the buoy location (43.6 ºN and 2.0 ºW) in radar and model grids are considered, and only the parts of years 2017 and 2018 which are present in the three sources (8319 hourly observations) are included.

Usage

```
data("Current")
```

Format

A list with two data frames:

`mod` a data frame including the data from both gridded datasets.

`ref` a data frame including the buoy data.

Each of these data frames includes 3 variables:

`mod` a factor defining if the data belong to the observed data (ref), to the radar (rad) or to the ocean modelling product (mod).

`U` zonal component of ocean current (m/s).

`V` meridional component of ocean current (m/s).

Source

The in-situ data belong to the Basque Meteorological Agency EUSKALMET and were downloaded from https://www.euskoos.eus. The Copernicus Marine Environment Monitoring Service (CMEMS) data can be retrieved from their data portal.

Examples

```
#-------------------------------------------------------
# Example 1: Figure 6 (left) from Sáenz et al. (2020)
#-------------------------------------------------------

# Load the data
data(Current)
```

```
# Parameters
Uxlim=c(-0.5,0.5)
Uylim=c(-0.5,0.5)
Uxlab<-"Ux (m/s)"
Uylab<-"Uy (m/s)"
plotmain<-"Surface current"
sfactor<-4
ref<-Current[["ref"]]
mod<-Current[["mod"]]

# Index
SailoR.Indices(ref,mod)

# Plot
SailoR.Plot(ref, mod, ColourList=NULL, sfactor, docenter=TRUE,
            Uxlim, Uylim, Uxlab, Uylab, plotmain, plotlegend=TRUE,
            plotRMSElegend=TRUE, plotscalelegend=TRUE,
            RMSE_legend_Rounding=2, RMSE_legend_units = " m/s",
            referenceName="Buoy")
```

`Dragonera` *Wind data*

Description

Contains the information of both zonal and meridional wind components measured in the Mediterranean sea by the in-situ buoy Dragonera, operated by the Spanish State Ports Authority - Puertos del Estado. Data from two WRF downscaling experiments (including the 3DVAR data assimilation every 3 hours - D, and not including it - N), ERA-Interim reanalysis and remote sensing data from the second version of the Cross-Calibrated Multi-platform (CCMPv2) product are also included. Only the closest spatial grid points to the buoy location (39.56 ºN, 2.10 ºE) in the gridded products are considered. The data cover the period 2009-2014 (Ulazia et al., 2017).

Usage

```
data("Dragonera")
```

Format

A list with two data frames:

`ref` a data frame including the buoy data.

`mod` a data frame including the data from gridded datasets.

Each of these data frames includes 3 variables:

`mod` a factor defining if the data belong to the CCMPv2 product (CCMP_SAT), to the reanalysis ERA-Interim (rad) or to any of the WRF downscaling experiments (WRF_D or WRF_N).

`U` zonal component of wind (m/s).

`V` meridional component of wind (m/s).

Source

Both WRF downscaling experiments were created by the authors in previous studies. The in-situ data can be retrieved from https://www.puertos.es/servicios/oceanografia. The CCMPv2 data can be retrieved from https://www.remss.com/measurements/ccmp/.

References

A. Ulazia, J. Sáenz, G. Ibarra-Berastegi, S. J. González-Rojí and S. Carreno-Madinabeitia (2017) Using 3DVAR data assimilation to measure offshore wind energy potential at different turbine heights in the West Mediterranean. Applied Energy, 208:1232-1245, doi:10.1016/j.apenergy.2017.09.030.

Examples

```
# Load the data
data(Dragonera)

# Parameters

Uxlab<-"Ux (m/s)"
Uylab<-"Vy (m/s)"
ref<-Dragonera[["ref"]]
mod<-Dragonera[["mod"]]

# Index
SailoR.Indices(ref,mod)

#-------------------------------------------------------
# Example 1: Figure 5 (left) from Sáenz et al. (2020)
#-------------------------------------------------------

Uxlim=c(-10,10)
Uylim=c(-10,10)
sfactor<-1
plotmain<-"Dragonera U10/V10"
p <- SailoR.Plot(ref,mod,ColourList=NULL,sfactor,docenter=TRUE,
                 Uxlim,Uylim,Uxlab,Uylab,plotmain,plotlegend=TRUE,
                 plotRMSElegend=TRUE,
                 plotscalelegend=TRUE,
                 RMSE_legend_Rounding=1,RMSE_legend_units = " m/s",
                 referenceName="Reference")

# Add segments
segments(-1,-1, x1 = -1, y1 = 1,col="brown",lwd=4)
segments(-1,-1, x1 = 1, y1 = -1,col="brown",lwd=4)
segments(1,-1, x1 = 1, y1 = 1,col="brown",lwd=4)
segments(1,1, x1 = -1, y1 = 1,col="brown",lwd=4)



#-------------------------------------------------------
# Example 2: Figure 5 (right) from Sáenz et al. (2020)
#-------------------------------------------------------
```

```
Uxlim=c(-1,1)
Uylim=c(-1,1)
sfactor<-0.025
plotmain<-"Dragonera U10/V10 (scaled)"
SailoR.Plot(ref,mod,ColourList=NULL,sfactor,docenter=FALSE,
            Uxlim,Uylim,Uxlab,Uylab,plotmain,plotlegend=TRUE,
            plotRMSElegend=TRUE,
            plotscalelegend=TRUE,
            RMSE_legend_Rounding=1,RMSE_legend_units = " m/s",
            referenceName="Reference")
```

`Ensembles` *Multimodel Ensembles*

Description

Contains the information of both components of surface wind over the Southern Hemisphere during the period 1979-2015 from the historical forcing experiment of CMIP5 repository. Six ensemble members from the IPSL model, five from the MIROC model and four from the HGEM are included. Additionally, data from the reanalysis ERA5 is also included. All of them have been bilinearly interpolated to a common 1.25º x 1º regular grid.

Usage

```
data("Ensembles")
```

Format

A list with 16 data frames:

`ERA5` a data frame including the data from reanalysis ERA5.

`IPSLr1` first realization of the IPSL model.

`IPSLr2` second realization of the IPSL model.

`IPSLr3` third realization of the IPSL model.

`IPSLr4` fourth realization of the IPSL model.

`IPSLr5` fifth realization of the IPSL model.

`IPSLr6` sixth realization of the IPSL model.

`MIROCr1` first realization of the MIROC model.

`MIROCr2` second realization of the MIROC model.

`MIROCr3` third realization of the MIROC model.

`MIROCr4` fourth realization of the MIROC model.

`MIROCr5` fifth realization of the MIROC model.

`HGEMr1` first realization of the HadGEM-ES model.

`HGEMr2` second realization of the HadGEM-ES model.

`HGEMr3` third realization of the HadGEM-ES model.

`HGEMr4` fourth realization of the HadGEM-ES model.

Each of these data frames includes 4 variables:

`V1` a vector with longitude data.

`V2` a vector with latitude data.

`V3` zonal component of surface wind (m/s).

`V4` meridional component of surface wind (m/s).

References

W. J. Collins, N. Bellouin, M. Doutriaux-Boucher, N. Gedney, P. Halloran, T. Hinton, ... and G. Martin (2011). Development and evaluation of an Earth-System model–HadGEM2. Geosci. Model Dev. Discuss, 4(2), 997-1062.

J. L. Dufresne, M. A. Foujols, S. Denvil, A. Caubel, O. Marti, O. Aumont, ... and S. Bony (2013). Climate change projections using the IPSL-CM5 Earth System Model: from CMIP3 to CMIP5. Climate Dynamics, 40(9-10), 2123-2165.

H. Hersbach, P. de Rosnay, B. Bell, D. Schepers, A. Simmons et al. (2018). Operational global reanalysis: progress, future directions and synergies with NWP. ECMWF Report.

M. Watanabe, T. Suzuki, R. O'ishi, Y. Komuro, S. Watanabe, S. Emori, ... and K. Takata (2010). Improved climate simulation by MIROC5: mean states, variability, and climate sensitivity. Journal of Climate, 23(23), 6312-6335.

Examples

```
# Load the data
data(Ensembles)

deg2rad<-function(d){
  return(d*pi/180.)
}

# Get area factors from latitudes
Ws<-sqrt(cos(deg2rad(Ensembles$ERA5$V2)))

# Create a ref/ mod object
# ERA5$V3 is zonal component and ERA5$V4 is meridional component
ref<-data.frame("ERA5",Ensembles$ERA5$V3*Ws,Ensembles$ERA5$V4*Ws)
names(ref)<-c("mod","U","V")

# Create an object for the model runs

modA<-data.frame(mod=rep("MIROCr1",nrow(Ensembles$MIROCr1)),
                            U=Ensembles$MIROCr1$V3*Ws,V=Ensembles$MIROCr1$V4*Ws)
modB<-data.frame(mod=rep("MIROC",nrow(Ensembles$MIROCr1)),
                            U=Ensembles$MIROCr1$V3*Ws,V=Ensembles$MIROCr1$V4*Ws)
```

```
# In modA every run of the ensemble is considered individually
modA<-rbind(modA,data.frame(mod=rep("MIROCr2",nrow(Ensembles$MIROCr2)),
                U=Ensembles$MIROCr2$V3*Ws,V=Ensembles$MIROCr2$V4*Ws))
# In modB all the members of the ensemble are taken as a single
# model run (too long, so that later on, observations will have to be repeated)
modB<-rbind(modB,data.frame(mod=rep("MIROC",nrow(Ensembles$MIROCr2)),
                U=Ensembles$MIROCr2$V3*Ws,V=Ensembles$MIROCr2$V4*Ws))

modA<-rbind(modA,data.frame(mod=rep("MIROCr3",nrow(Ensembles$MIROCr3)),
                            U=Ensembles$MIROCr3$V3*Ws,V=Ensembles$MIROCr3$V4*Ws))
modB<-rbind(modB,data.frame(mod=rep("MIROC",nrow(Ensembles$MIROCr3)),
                            U=Ensembles$MIROCr3$V3*Ws,V=Ensembles$MIROCr3$V4*Ws))

modA<-rbind(modA,data.frame(mod=rep("MIROCr4",nrow(Ensembles$MIROCr4)),
                            U=Ensembles$MIROCr4$V3*Ws,V=Ensembles$MIROCr4$V4*Ws))
modB<-rbind(modB,data.frame(mod=rep("MIROC",nrow(Ensembles$MIROCr4)),
                            U=Ensembles$MIROCr4$V3*Ws,V=Ensembles$MIROCr4$V4*Ws))

modA<-rbind(modA,data.frame(mod=rep("MIROCr5",nrow(Ensembles$MIROCr5)),
                            U=Ensembles$MIROCr5$V3*Ws,V=Ensembles$MIROCr5$V4*Ws))
modB<-rbind(modB,data.frame(mod=rep("MIROC",nrow(Ensembles$MIROCr5)),
                            U=Ensembles$MIROCr5$V3*Ws,V=Ensembles$MIROCr5$V4*Ws))

modA<-rbind(modA,data.frame(mod=rep("IPSLr1",nrow(Ensembles$IPSLr1)),
                            U=Ensembles$IPSLr1$V3*Ws,V=Ensembles$IPSLr1$V4*Ws))
modB<-rbind(modB,data.frame(mod=rep("IPSL",nrow(Ensembles$IPSLr1)),
                            U=Ensembles$IPSLr1$V3*Ws,V=Ensembles$IPSLr1$V4*Ws))

modA<-rbind(modA,data.frame(mod=rep("IPSLr2",nrow(Ensembles$IPSLr2)),
                            U=Ensembles$IPSLr2$V3*Ws,V=Ensembles$IPSLr2$V4*Ws))
modB<-rbind(modB,data.frame(mod=rep("IPSL",nrow(Ensembles$IPSLr2)),
                            U=Ensembles$IPSLr2$V3*Ws,V=Ensembles$IPSLr2$V4*Ws))

modA<-rbind(modA,data.frame(mod=rep("IPSLr3",nrow(Ensembles$IPSLr3)),
                            U=Ensembles$IPSLr3$V3*Ws,V=Ensembles$IPSLr3$V4*Ws))
modB<-rbind(modB,data.frame(mod=rep("IPSL",nrow(Ensembles$IPSLr3)),
                            U=Ensembles$IPSLr3$V3*Ws,V=Ensembles$IPSLr3$V4*Ws))

modA<-rbind(modA,data.frame(mod=rep("IPSLr4",nrow(Ensembles$IPSLr4)),
                            U=Ensembles$IPSLr4$V3*Ws,V=Ensembles$IPSLr4$V4*Ws))
modB<-rbind(modB,data.frame(mod=rep("IPSL",nrow(Ensembles$IPSLr4)),
                            U=Ensembles$IPSLr4$V3*Ws,V=Ensembles$IPSLr4$V4*Ws))

modA<-rbind(modA,data.frame(mod=rep("IPSLr5",nrow(Ensembles$IPSLr5)),
                            U=Ensembles$IPSLr5$V3*Ws,V=Ensembles$IPSLr5$V4*Ws))
modB<-rbind(modB,data.frame(mod=rep("IPSL",nrow(Ensembles$IPSLr5)),
                            U=Ensembles$IPSLr5$V3*Ws,V=Ensembles$IPSLr5$V4*Ws))

modA<-rbind(modA,data.frame(mod=rep("IPSLr6",nrow(Ensembles$IPSLr6)),
                            U=Ensembles$IPSLr6$V3*Ws,V=Ensembles$IPSLr6$V4*Ws))
modB<-rbind(modB,data.frame(mod=rep("IPSL",nrow(Ensembles$IPSLr6)),
                            U=Ensembles$IPSLr6$V3*Ws,V=Ensembles$IPSLr6$V4*Ws))
```

```
modA<-rbind(modA,data.frame(mod=rep("HGEMr1",nrow(Ensembles$HGEMr1)),
                            U=Ensembles$HGEMr1$V3*Ws,V=Ensembles$HGEMr1$V4*Ws))
modB<-rbind(modB,data.frame(mod=rep("HGEM",nrow(Ensembles$HGEMr1)),
                            U=Ensembles$HGEMr1$V3*Ws,V=Ensembles$HGEMr1$V4*Ws))

modA<-rbind(modA,data.frame(mod=rep("HGEMr2",nrow(Ensembles$HGEMr2)),
                            U=Ensembles$HGEMr2$V3*Ws,V=Ensembles$HGEMr2$V4*Ws))
modB<-rbind(modB,data.frame(mod=rep("HGEM",nrow(Ensembles$HGEMr2)),
                            U=Ensembles$HGEMr2$V3*Ws,V=Ensembles$HGEMr2$V4*Ws))

modA<-rbind(modA,data.frame(mod=rep("HGEMr3",nrow(Ensembles$HGEMr3)),
                            U=Ensembles$HGEMr3$V3*Ws,V=Ensembles$HGEMr3$V4*Ws))
modB<-rbind(modB,data.frame(mod=rep("HGEM",nrow(Ensembles$HGEMr3)),
                            U=Ensembles$HGEMr3$V3*Ws,V=Ensembles$HGEMr3$V4*Ws))

modA<-rbind(modA,data.frame(mod=rep("HGEMr4",nrow(Ensembles$HGEMr4)),
                            U=Ensembles$HGEMr4$V3*Ws,V=Ensembles$HGEMr4$V4*Ws))
modB<-rbind(modB,data.frame(mod=rep("HGEM",nrow(Ensembles$HGEMr4)),
                            U=Ensembles$HGEMr4$V3*Ws,V=Ensembles$HGEMr4$V4*Ws))

CMIPA<-list(ref,modA)
names(CMIPA)<-c("ref","mod")
CMIPB<-list(ref,modB)
names(CMIPB)<-c("ref","mod")

#-------------------------------------------------------
# Example 1: Figure 8 (right) from Sáenz et al. (2020)
#-------------------------------------------------------

# Parameters
Uxlim=c(-1,1)
Uylim=c(-0.5,1.5)
Uxlab<-"Ux (m/s)"
Uylab<-"Uy (m/s)"
plotmain<-"CMIP5 samples vs ERA5"
sfactor<-0.2

ref<-CMIPB[["ref"]]
mod<-CMIPB[["mod"]]

sindex<-SailoR.Indices(ref, mod, Ensembles=TRUE)
SailoR.Table(sindex, round_digits = 3)


SailoR.Plot(ref, mod, ColourList=NULL, sfactor, docenter=TRUE,
            Uxlim, Uylim, Uxlab, Uylab, plotmain, plotlegend=TRUE,
            Ensembles=TRUE, plotRMSElegend=TRUE, plotscalelegend=TRUE,
            RMSE_legend_Rounding=2, RMSE_legend_units = " m/s",
            referenceName="ERA5")

#-------------------------------------------------------
# Example 2: Figure 9 (rigth) from Sáenz et al. (2020)
#-------------------------------------------------------
```

```
# Parameters
Uxlim=c(-0.15,0.25)
Uylim=c(0.19,0.61)
Uxlab<-"Ux (m/s)"
Uylab<-"Uy (m/s)"
plotmain<-"CMIP5 samples vs ERA5"
sfactor<-0.01
theColourList<-c("grey40","red","blue","darkolivegreen3",
                 "orange","seagreen3","gold",
                 "purple","pink","magenta",
                 "lightblue","gold4","darkslateblue",
                 "coral","green4","deeppink4")

ref<-CMIPA[["ref"]]
mod<-CMIPA[["mod"]]

sindex<-SailoR.Indices(ref,mod,Ensembles=TRUE)
ST<-SailoR.Table(sindex,round_digits = 1)

p<-SailoR.Plot(ref, mod, ColourList=theColourList, sfactor, docenter=FALSE,
            Uxlim, Uylim, Uxlab, Uylab, plotmain, plotlegend=FALSE,
            Ensembles=TRUE, plotRMSElegend=FALSE, plotscalelegend=TRUE,
            RMSE_legend_Rounding=2, RMSE_legend_units = " m/s",
            referenceName="ERA5")

#legend
legend(x="topright",
       as.character(ST$modelName),
       col=theColourList,lwd=2,
       lty=c("dashed",rep("solid",15)),cex=1,ncol=2)

#RMSElegend
legend(x="bottomright",
       paste(as.character(unique(mod$mod)),": ",ST$RMSE[-1],"m/s",sep=""),
       col=theColourList,
       pch=16,lwd=3,
       title=expression(bold("RMSE")),
       cex=0.9,ncol=3)
```

`EstacaDeBares` *Wave Energy Flux data*

Description

Contains the information of both components of wave energy flux measured by the buoy placed in Estaca de Bares and operated by the Spanish State Ports Authority - Puertos del Estado. Different statistical models were applied to this observed data: analogs, random forests, the combination of both of them, WAM model and the persistence (Ibarra-Berastegi et al., 2015 and 2016).

Usage

```
data("EstacaDeBares")
```

Format

A list with two data frames:

`ref` a data frame including the buoy data.

`mod` a data frame including the data from the statistical models.

Each of these data frames includes 4 variables:

`date` date information.

`mod` a factor defining if the data belong to the analogs method (analo), to random forests (rf), both techniques combined (analorf), WAM model (wam) or persistence (pers).

`U` zonal component of wave energy flux (kW/m).

`V` meridional component of wave energy flux (kW/m).

Source

The results from the statistical methods were obtained by the authors in previous studies. The in-situ data can be retrieved from https://www.puertos.es/servicios/oceanografia.

References

G. Ibarra-Berastegi, J. Sáenz, G. Esnaola, A. Ezcurra and A. Ulazia (2015) Short-term forecasting of the wave energy flux: Analogues, random forests, and physics-based models. Ocean Engineering, 104:530-539, doi:10.1016/j.oceaneng.2015.05.038.

G. Ibarra-Berastegi, J. Sáenz, G. Esnaola, A. Ezcurra, A. Ulazia, N. Rojo and G. Gallastegui (2016) Wave Energy Forecasting at Three Coastal Buoys in the Bay of Biscay IEEE JOURNAL OF OCEANIC ENGINEERING, 41:923-929, doi:10.1109/JOE.2016.2529400.

Examples

```
#-----------------------------------------------------------
# Example 1
#-----------------------------------------------------------

# Load the data
data(EstacaDeBares)

# Parameters
Uxlim=c(0,25)
Uylim=c(0,25)
Uxlab<-" "
Uylab<-" "
plotmain<-"Wave Energy Flux. Estaca De Bares."
sfactor<-0.25
plotRMSE<-TRUE
ref<-EstacaDeBares[["ref"]][,c("mod","U","V")]
```

```
mod<-EstacaDeBares[["mod"]][,c("mod","U","V")]

# Index
SailoR.Indices(ref,mod)

# Plot
SailoR.Plot(ref, mod, ColourList=NULL, sfactor, docenter=TRUE,
            Uxlim, Uylim, Uxlab, Uylab, plotmain, plotlegend=TRUE,
            plotRMSElegend=TRUE, plotscalelegend=TRUE,
            RMSE_legend_Rounding=0,RMSE_legend_units = " W/m",
            referenceName="Buoy")
```

`Reanalysis` *Surface Wind from different Reanalyses*

Description

Contains the information of both components of monthly averaged surface wind in January over the Northern Hemisphere as included in different state-of-the-art reanalyses during the period 2011-2018. ERA5 reanalysis was defined as the reference, and the remaining gridded products were included in the model matrix. Particularly, the original NCEP/NCAR first generation reanalysis, MERRA2, CFSv2 and ERA-Interim. All of them have been bilinearly interpolated to NCEP/NCAR original grid.

Usage

```
data("Reanalysis")
```

Format

A list with two matrices:

`ref` a matrix including the data from reanalysis ERA5.

`mod` a matrix including the data from other reanalyses.

Each of these data frames includes 5 variables:

`V1` a vector with longitude data.

`V2` a vector with latitude data.

`V3` a factor defining if the data belong to the reanalysis CFSv2 (cfsv2), ERA-Interim (ei), MERRA2 (merra2) or NCEP/NCAR reanalysis (nnra).

`V4` zonal component of surface wind (m/s).

`V5` meridional component of surface wind (m/s).

Examples

```
# Load the data
data(Reanalysis)

# Parameters
Uxlim=c(-0.5,1)
Uylim=c(-1,0.5)
Uxlab<-"Ux (m/s)"
Uylab<-"Uy (m/s)"
plotmain<-"Reanalyses"
sfactor<-0.15

# Create ref, mod objects

weightWithLat=TRUE

if(weightWithLat){
  dlats=as.numeric(Reanalysis[["ref"]][,2])
  rlats=pi*dlats/180.
  lweights=sqrt(cos(rlats))

  ref<-data.frame(Reanalysis[["ref"]][,3],
                  lweights*as.numeric(Reanalysis[["ref"]][,4]),
                  lweights*as.numeric(Reanalysis[["ref"]][,5]))
}else{

  ref<-data.frame(Reanalysis[["ref"]][,3],
                  as.numeric(Reanalysis[["ref"]][,4]),
                  as.numeric(Reanalysis[["ref"]][,5]))

}
names(ref)<-c("mod","U","V")

mod<-data.frame(Reanalysis[["mod"]][,3],
                as.numeric(Reanalysis[["mod"]][,4]),
                as.numeric(Reanalysis[["mod"]][,5]))
names(mod)<-c("mod","U","V")

# Sailors

#-------------------------------------------------------
# Example 1: Figure 7 (left) from Sáenz et al. (2020)
#-------------------------------------------------------

SailoR.Plot(ref, mod, ColourList=NULL, sfactor, docenter=FALSE,
            Uxlim, Uylim, Uxlab, Uylab, plotmain, plotlegend=TRUE,
            plotRMSElegend=TRUE, plotscalelegend=TRUE,
            RMSE_legend_Rounding=1, RMSE_legend_units = " m/s",
            referenceName="ERA5")

#-------------------------------------------------------
# Example 2: Figure 7 (right) from Sáenz et al. (2020)
```

```
#-----------------------------------------------------

SailoR.Plot(ref, mod, ColourList=NULL, sfactor, docenter=TRUE,
            Uxlim, Uylim, Uxlab, Uylab, plotmain, plotlegend=TRUE,
            plotRMSElegend=TRUE, plotscalelegend=TRUE,
            RMSE_legend_Rounding=1, RMSE_legend_units = " m/s",
            referenceName="ERA5")
```

`SailoR.Indices` *Calculation of indices for Sailor Diagram*

Description

This function returns a data frame with all the statistics used by the Sailor diagram for the set of different models in data frame `mod` with the reference in data frame `ref`.

Usage

```
SailoR.Indices(ref, mod, Ensembles = FALSE)
```

Arguments

`ref`	A data frame with the reference observations. It should have 3 columns: name, zonal component and meridional component.
`mod`	A data frame with the model data. It should have 3 columns: name, zonal component and meridional component. More than one model can be included under different names. Additionally, `nrows(mod)` must be an integer multiple of `nrows(ref)`.
`Ensembles`	If `Ensembles` is set to `TRUE`, M different models included in the data frame `mod` will be taken as a single model and the reference model is repeated M times for the calculation of the Sailor diagram.

Value

The result is a data frame with a summary of the statistics obtained:

`meanU`	mean of the U dataset.
`meanV`	mean of the V dataset.
`RMSE`	root mean square error between U and V.
`sdUx`	standard deviation of zonal component (U).
`sdUy`	standard deviation of meridional component (U).
`sdVx`	standard deviation of zonal component (V).
`sdVy`	standard deviation of meridional component (V).
`Eu`	EOF matrix for the U dataset.

Ev	EOF matrix for the V dataset.
Sigmau	matrix with standard deviations of U (2x2 matrix).
Sigmav	matrix with standard deviations of V (2x2 matrix).
TotVarU	total variance of the U dataset.
TotVarV	total variance of the V dataset.
thetau	angle of EOF1 (U) with zonal axis.
thetav	angle of EOF1 (V) with meridional axis.
thetavu	rotation angle from U EOFs to V EOFs.
R2vec	vector correlation squared in 2D as defined by Breaker, Gemmill and Crossby (1994).
Rvu	rotation matrix to express Ev as a rotation from Eu.
EccentricityU	eccentricity of the ellipses from the reference dataset.
EccentricityV	eccentricity of the ellipses from the model dataset.
congruenceEOF	congruence coefficients (absolute value) for EOFs.

References

L. C. Breaker, W. H. Gemmill and D. S. Crosby (1994). The application of a technique for vector correlation to problems in meteorology and oceanography. Journal of Applied Meteorology, 33(11), 1354-1365.

Examples

```
#-------------------------------------------------------
# Example 1
#-------------------------------------------------------

# Load the ocean current data
data("Current")

# Calculate the indices
ref<-Current[["ref"]]
mod<-Current[["mod"]]

SailoR.Indices(ref, mod)
```

SailoR.Plot *Sailor Diagram*

Description

This function generates the Sailor diagram. It calls `SailoR.Indices()` internally.

Usage

```
SailoR.Plot(ref, mod, ColourList = NULL, sfactor = 1, docenter = FALSE,
xlim = NULL, ylim = NULL, xlab = NULL, ylab = NULL,
plotmain = NULL, plotlegend = TRUE, plotRMSElegend = TRUE,
plotscalelegend=TRUE, Ensembles = FALSE, RMSE_legend_units = "",
RMSE_legend_Rounding = 2, referenceName = NULL,
linestype = NULL, bias_pch = NULL)
```

Arguments

ref	a data frame with the reference observations. It should have 3 columns: name, zonal component and meridional component.
mod	a data frame with the model data. It should have 3 columns: name, zonal component and meridional component. More than one model can be included by using different names.
ColourList	a vector with all the colors you want to use for plotting the models. Darkgray will always be used for reference data. If NULL, the default colours are used: c("gray40", "red", "blue", "darkolivegreen3", "orange", "seagreen3", "gold", "purple", "pink").
sfactor	a value with the scale factor to be applied to the ellipses of each model.
docenter	if all the ellipses should be plotted over the reference one, this argument should be TRUE.
xlim	X axis limit for the plot.
ylim	Y axis limit for the plot.
xlab	a title for the X axis.
ylab	a title for the Y axis.
plotmain	an overall title for the plot.
plotlegend	if a default legend should be added to the plot, this argument should be TRUE.
plotRMSElegend	if a legend with a summary of the RMSE values should be added to the plot, this argument should be TRUE.
plotscalelegend	if a legend with the scaled factor should be added to the plot, this argument should be TRUE.
Ensembles	if Ensembles is set to TRUE, M different models included in the data frame mod will be taken as a single model and the reference model is repeated M times for the calculation of the Sailor diagram.
RMSE_legend_units	units to be written in the RMSE legend.
RMSE_legend_Rounding	number of decimals to be kept in the RMSE legend.
referenceName	external name provided for the reference dataset for the legend.
linestype	a vector with the lines types for the model data. The "dashed" option is reserved for the reference. The other options are "solid", "dotted", "dotdash", "longdash" or "twodash".

bias_pch — a vector with the pch types for the bias symbol. The default opcion provides a circle when docenter is TRUE and a point when it is FALSE.

Value

The result is a plot object.

See Also

SailoR.Indices

Examples

```
#-----------------------------------------------------------
# Example 1: Figure 6 (right) from Sáenz et al. (2020)
#-----------------------------------------------------------

# Load the Vertically Integrated Water Vapor Transport Data included in the package
data(WRF)

# Parameters
Uxlim=c(60,110)
Uylim=c(0,50)
Uxlab<-"Qx (kg/m/s)"
Uylab<-"Qy (kg/m/s)"
plotmain<-"Water vapour transport"
sfactor<-0.1
ref<-WRF[["ref"]][,c("mod","U","V")]
mod<-WRF[["mod"]][,c("mod","U","V")]

isBad=((is.na(ref$U)) |(is.na(ref$V)))
isOK=(!isBad)
ref<-ref[isOK,]

nmod1<-which(mod$mod=="WrfN")
mod1<-mod[nmod1,][isOK,]
nmod2<-which(mod$mod=="WrfD")
mod2<-mod[nmod2,][isOK,]
nmod3<-which(mod$mod=="ERAI")
mod3<-mod[nmod3,][isOK,]

mod<-rbind(mod1,mod2,mod3)

# Index
sIWRF=SailoR.Indices(ref,mod)

# Index table
sIWRF.table<-SailoR.Table(sIWRF)

# Plot
SailoR.Plot(ref,mod,ColourList=NULL,sfactor,docenter=TRUE,
            Uxlim, Uylim, Uxlab, Uylab, plotmain, plotlegend=TRUE,
```

```
plotRMSElegend=TRUE, RMSE_legend_Rounding=0,
RMSE_legend_units=" kg/m/s", referenceName="Sounding",
,linestype=c("solid", "dotted", "twodash"))
```

SailoR.Table *Summary of indices calculated for the Sailor diagrams*

Description

This function reads the output from `SailoR.Indices()` and creates a summary data frame of the indices calculated for the Sailor diagram. This can easily be converted to a LaTeX or other formats with external packages (`'xtable'`, for instance).

Usage

```
SailoR.Table(UV, round_digits = 2)
```

Arguments

`UV`	a list containing the output obtained directly from `SailoR.Indices()` function.
`round_digits`	number of decimals to be kept in the summary table.

Value

The result is a data frame with the information from the `SailoR.Indices()` function in tabular format. Particularly, the indices stated in the table are the following ones:

`modelName`	the names of the models datasets.
`sdUx`	standard deviation of zonal component (U).
`sdUy`	standard deviation of meridional component (U).
`sdVx`	standard deviation of zonal component (V).
`sdVy`	standard deviation of meridional component (V).
`Sigmax`	a scalar representing the semi-major axis of the ellipse (standard deviation of PC1).
`Sigmay`	a scalar representing the semi-minor axis of the ellipse (standard deviation of PC2).
`thetau`	angle of EOF1 (U) with zonal axis.
`thetav`	angle of EOF1 (V) with meridional axis.
`thetavu`	rotation angle from U EOFs to V EOFs.
`R2vec`	vector correlation squared in 2D as defined by Breaker, Gemmill and Crossby (1994).
`biasMag`	magnitude of the bias vector.
`RMSE`	root mean square error between U and V.
`Eccentricity`	eccentricity of the ellipses from the reference and models dataset.
`congruenceEOF1`	congruence coefficients (absolute value) for EOF1.

Examples

```
#-------------------------------------------------------
# Example 1
#-------------------------------------------------------
# Load the ocean current data
data("Current")

# Calculate the statistics
ref<-Current[["ref"]]
mod<-Current[["mod"]]
UV <- SailoR.Indices(ref,mod)

# Obtain the summary of the results
SailoR.Table(UV)

#-------------------------------------------------------
# Example 2: Table 2 from Sáenz et al. (2020)
#-------------------------------------------------------
# Load Synthetic data
data(Synthetic)

ref<-Synthetic[["ref"]]
mod<-Synthetic[["mod"]]

# Calculate the statistics
UVtable=SailoR.Table(UV,round_digits = 4)
TotVar<-c((UVtable$sdUx^2+UVtable$sdUy^2)[1],
          (UVtable$sdVx^2+UVtable$sdVy^2)[-1])
TotSigma<-UVtable$Sigmax^2+UVtable$Sigmay^2
UVtable<-cbind(UVtable[,1],TotVar,TotSigma,UVtable[,8:15])

UVtable<-round(UVtable[,-1],2)

#If you want to print it as a LaTeX table:
#library(xtable)
#oInfo<-xtable(UVtable)
#print(oInfo,type="latex",file="table.tex")
```

`Synthetic` *Ocean Current Data*

Description

This dataset contains a one-year long time-series of hourly zonal and meridional wind components in the extratropical Pacific from ERA5. From this dataset (Reference), different synthetic datasets with individual sources of error artificially added to them have been produced. In the first case (MOD1), a constant bias has been added. Next, the error is produced (MOD2) by rotating the zonal and meridional components by 30 degrees counterclockwise (thus also inducing a bias due

to the rotation of the mean vector). An unphysical source of error is added in MOD3 by randomly resampling the dataset in order to break the original correlation of the vectors while keeping the bias and EOFs at their original values. Finally, the original data has been multiply by a constant in order to change the variance of the data (MOD4), although the scaling produces a change in the bias, too. The main objective of adding this synthetic dataset is to show the response of the Sailor diagram to different kinds of errors.

Usage

```
data("Synthetic")
```

Format

A list with two data frames:

`mod` a data frame including the wind data from ERA5.

`ref` a data frame including the synthetic datasets.

Each of these data frames includes 3 variables:

`mod` a factor defining if the data belong to the ERA5 wind data (REF), or to the synthetic datasets MOD1, MOD2, MOD3 or MOD4.

`U` zonal component of wind speed (m/s).

`V` meridional component of wind speed (m/s).

Examples

```
# Load the data
data(Synthetic)

Uxlim=c(0,15)
Uylim=c(-25,10)
Uxlab<-"Ux (m/s)"
Uylab<-"Uy (m/s)"
plotmain<-"Reference and synthetic models"
sfactor<-1
ref<-Synthetic[["ref"]]
mod<-Synthetic[["mod"]]

#-------------------------------------------------------
# Example 1: Figure 4 (left) from Sáenz et al. (2020)
#-------------------------------------------------------

SailoR.Plot(ref, mod, ColourList=NULL, sfactor, docenter=FALSE,
            Uxlim, Uylim, Uxlab, Uylab, plotmain, plotlegend=TRUE,
            Ensembles=TRUE, plotRMSElegend=TRUE,
            plotscalelegend=TRUE, RMSE_legend_Rounding=2,
            RMSE_legend_units = " m/s")

#-------------------------------------------------------
# Example 2: Figure 4 (right) from Sáenz et al. (2020)
```

```
#-----------------------------------------------------

SailoR.Plot(ref, mod, ColourList=NULL, sfactor, docenter=TRUE,
            Uxlim, Uylim, Uxlab, Uylab, plotmain, plotlegend=TRUE,
            Ensembles=TRUE, plotRMSElegend=TRUE,
            plotscalelegend=TRUE, RMSE_legend_Rounding=2,
            RMSE_legend_units = " m/s")
```

`UVError` *Calculation of the elements of the 2D MSE terms*

Description

This function calculates the different terms in the 2D MSE equation, the rotation of EOFs and so on for a unique given model. This function is internally used by `SailoR.Indices()` and external users are not expected to use it.

Usage

```
UVError(U, V, Ensembles = FALSE)
```

Arguments

`U`	a vector including the zonal and meridional components of the reference dataset.
`V`	a vector including the zonal and meridional components of the model being compared.
`Ensembles`	if `Ensembles` is set to `TRUE`, M different models included in the data frame `mod` will be taken as a single model and the reference model is repeated M times for the calculation of the Sailor diagram.

Value

The result is a list with a summary of the statistics obtained:

`meanU`	mean of the U dataset.
`meanV`	mean of the V dataset.
`TotVarU`	total variance of the U dataset.
`TotVarV`	total variance of the V dataset.
`Eu`	EOF matrix for the U dataset.
`Ev`	EOF matrix for the V dataset.
`Rvu`	rotation matrix to express Ev as a rotation from Eu.
`Sigmau`	matrix with standard deviations of U (2x2 matrix).
`Sigmav`	matrix with standard deviations of V (2x2 matrix).
`sdUx`	standard deviation of zonal component (U).

sdUy	standard deviation of meridional component (U).
sdVx	standard deviation of zonal component (V).
sdVy	standard deviation of meridional component (V).
thetau	angle of EOF1 (U) with zonal axis.
thetav	angle of EOF1 (V) with meridional axis.
thetavu	rotation angle from U EOFs to V EOFs.
RMSE	root mean square error between U and V.
R2vec	vector correlation squared in 2D as defined by Breaker, Gemmill and Crossby (1994).
EccentricityU	eccentricity of the ellipses from the reference dataset.
EccentricityV	eccentricity of the ellipses from the model dataset.
congruenceEOF	congruence coefficients (absolute value) for EOFs.

References

L. C. Breaker, W. H. Gemmill, and D. S. Crosby (1994). The application of a technique for vector correlation to problems in meteorology and oceanography. Journal of Applied Meteorology, 33(11), 1354-1365.

See Also

SailoR.Indices

Examples

```
# No examples are given as external users are not expected to use it
```

WRF *Vertically Integrated Water Vapor Transport Data*

Description

Contains the information of both components of the vertically integrated water vapor transports calculated over A Coruña (43.6 ºN, -8.41 ºE) in the Iberian Peninsula. This quantity is obtained after integrating the product of moisture and both components of wind across the whole vertical dimension of the atmosphere. To do so, moisture and wind components measured by radiosondes were retrieved from the server of the University of Wyoming, and used for its calculation. Data from two WRF downscaling experiments (including the 3DVAR data assimilation every 3 hours - D experiment, and not including it - N experiment) and ERA-Interim data were also used for its calculation. In these cases, only the closest spatial grid points to the station in the gridded products are considered. 12-hourly data from period 2010-2014 are included with missing soundings removed (González-Rojí et al., 2018 and 2019).

Usage

```
data("WRF")
```

Format

A list with two data frames:

`ref` a data frame including the vertically integrated water vapor transports calculated by means of radiosonde data.

`mod` a data frame including the data from gridded datasets.

Each of these data frames includes 4 variables:

`date` date information.

`mod` a factor defining if the values were calculated by means of reanalysis ERA-Interim data (ERAI) or data from WRF downscaling experiments (WrfD or WrfN).

`U` zonal component of vertically integrated water vapor transport (kg/m/s).

`V` meridional component of vertically integrated water vapor transport (kg/m/s).

Source

Both downscaling experiments were created by the authors in previous studies. ERA-Interim were retrieved from the MARS repository at ECMWF.

References

S. J. González-Rojí, J. Sáenz, G. Ibarra-Berastegi and J. Díaz-Argandoña (2018) Moisture balance over the Iberian Peninsula according to a regional climate model. The impact of 3DVAR data assimilation. Journal of Geophysical Research-Atmospheres, 123:708-729, doi:10.1002/2017JD027511

S. J. González-Rojí, R. L. Wilby, J. Sáenz and G. Ibarra-Berastegi (2019) Harmonized evaluation of daily precipitation downscaled using SDSM and WRF+WRFDA models over the Iberian Peninsula Climate Dynamics, 53:1413-1433, doi:10.1007/s00382019046739

Examples

```
#-----------------------------------------------------
# Example 1: Figure 6 (right) from Sáenz et al. (2020)
#-----------------------------------------------------

# Load the data
data("WRF")

# Parameters
Uxlim=c(60,110)
Uylim=c(0,50)
Uxlab<-"Qx (kg/m/s)"
Uylab<-"Qy (kg/m/s)"
plotmain<-"Water vapour transport"
sfactor<-0.1
ref<-WRF[["ref"]][,c("mod","U","V")]
```

```
mod<-WRF[["mod"]][,c("mod","U","V")]

isBad=((is.na(ref$U)) |(is.na(ref$V)))
isOK=(!isBad)
ref<-ref[isOK,]

nmod1<-which(mod$mod=="WrfN")
mod1<-mod[nmod1,][isOK,]
nmod2<-which(mod$mod=="WrfD")
mod2<-mod[nmod2,][isOK,]
nmod3<-which(mod$mod=="ERAI")
mod3<-mod[nmod3,][isOK,]

mod<-rbind(mod1,mod2,mod3)

# Index
sIWRF=SailoR.Indices(ref,mod)

# Index table
sIWRF.table<-SailoR.Table(sIWRF)

# plot

SailoR.Plot(ref, mod, ColourList=NULL, sfactor, docenter=TRUE,
            Uxlim, Uylim, Uxlab, Uylab, plotmain, plotlegend=TRUE,
            plotRMSElegend=TRUE, plotscalelegend=TRUE,
            RMSE_legend_Rounding=0, RMSE_legend_units = " kg/m/s",
            referenceName="Sounding")
```

Index

∗ **Datasets**
Current, 4
Dragonera, 5
Ensembles, 7
EstacaDeBares, 11
Reanalysis, 13
Synthetic, 20
WRF, 23
∗ **Functions**
SailoR.Indices, 15
SailoR.Plot, 16
SailoR.Table, 19
UVError, 22
∗ **Package**
SailoR-package, 2

Current, 4

Dragonera, 5

Ensembles, 7
EstacaDeBares, 11

Reanalysis, 13

SailoR *(*SailoR-package*)*, 2
SailoR-package, 2
SailoR.Indices, 15, *18*, *23*
SailoR.Plot, 16
SailoR.Table, 19
Synthetic, 20

UVError, *3*, 22

WRF, 23