Package ‘phytotools’

February 20, 2015

Type Package

Title Phytoplankton Production Tools

Version 1.0

Date 2015-02-10

Author Greg M. Silsbe, Sairah Y. Malkin

Maintainer Greg Silsbe <gsilsbe@gmail.com>

Depends insol, FME

Description Fits PE and RLC data to one of a four published PE models.
Simulates incident irradiance as a function of time and space.
Calculates phytoplankton production by transposing modeled PE or RLC data
to a water column with a user-defined theoretical in-situ irradiance field.

License GPL-2

NeedsCompilation no

Repository CRAN

Date/Publication 2015-02-14 01:08:08

R topics documented:

phytotools-package *Phytoplankton Production Tools*

Description

Fits PE and RLC data to one of a four published PE models. Simulates incident irradiance as a function of time and space. Calculates phytoplankton production by transposing modeled PE or RLC data to a water column with a user-defined theoretical in-situ irradiance field.

Author(s)

Greg M. Silsbe Sairah Y. Malkin

Maintainer: Greg Silsbe <gsilsbe@gmail.com>

References

Eilers, P.H.C. and Peeters, J.C.H. 1988 A model for the relationship between light intensity and the rate of photosynthesis in phytoplankton. *Ecological Modeling*. **42**, 199–215.

Fee, E.J. 1990. Computer programs for calculating in-situ phytoplankton photosynthesis. *Canadian technical report for fisheries and aquatic sciences*. **No. 1740**.

Hofierka, J., and Suri, M. 2002 The solar radiation model for Open source GIS: Implementation and applications. *Proceedings of the source GIS-GRASS users conference 2002*.

Jassby, A.D.and Platt, T. 1976 Mathematical formulation of the relationship between photosynthesis and light for phytoplankton. *Limnology and Oceanography*. **21**, 540–547.

Kirk, J.T.O., 2011 Light and photosynthesis in aquatic environments. Cambridge Press.

Platt, T., Gallegos, C.L. and Harrison, W.G. 1980 Photoinhibition and photosynthesis in natural assemblages of marine phytoplankton. *Journal of Marine Research*. **38**, 687–701.

Silsbe, G.M., and Kromkamp, J.C. 2012 Modeling the irradiance dependency of the quantum efficiency of photosynthesis. *Limnology and Oceanography: Methods*. **10**, 642–652.

Soetaert, K. and Petzoldt, T. 2010 Inverse Modelling, Sensitivity and Monte Carlo Analysis in R Using Package FME. *Journal of Statistical Software*. **33**, 1?28.

Webb, W.L., Newton, M., and Starr, D. 1974 Carbon dioxide exchange of *Alnus rubra*: A mathematical model. *Oecologia*. **17**, 281–291.

fitEP *Fit PE or RLC data to Eilers and Peeters 1988*

Description

Calculates photosynthetic-irradiance (PE) parameters (alpha, eopt, ps) and fit statistics for PE or RLC data using the model of Eilers and Peeters 1988.

Usage

```
fitEP(x, y, normalize = FALSE, lowerlim = c(0, 0, 0), upperlim = c(100, 2000, 2000),
fitmethod=c("Nelder-Mead"))
```

Arguments

`x`	PAR data. Units of umol m-2 s-1
`y`	Photosynthetic rate or PSII quantum efficiency.
`normalize`	Boolean. Default is FALSE. Set to TRUE if fitting PSII quantum efficiency. See Details.
`lowerlim`	Lower limits of parameter estimates (alpha, eopt, ps).
`upperlim`	Upper limits of parameter estimates (alpha, eopt, ps).
`fitmethod`	The method to be used, one of "Marq", "Port", "Newton", "Nelder-Mead", "BFGS", "CG", "L-BFGS-B", "SANN", "Pseudo". Default is "Nelder-Mead" - see details.

Details

This function passes the data to the function modFIT in the package FME that, through minimization via the specified 'fitmethod' algorithm, determines the optimal model parameters. See the help on modFit algorithms. "Nelder-Mead" is fast and works well for two parameter models, "SANN" is slow and works well for three parameter models.

If normalize is set to FALSE, then data is fit to the equation:

$$y = \frac{x}{x^2 \frac{1}{alpha \times epot^2} + \frac{x}{ps} - \frac{2x}{alpha \times eopt} + \frac{1}{alpha}}$$

If normalize is set to TRUE, then data is fit to the same equation but normalized to irradiance:

$$y = \frac{1}{x^2 \frac{1}{alpha \times epot^2} + \frac{x}{ps} - \frac{2x}{alpha \times eopt} + \frac{1}{alpha}}$$

Fitting a E normalized PE model is useful for modeling the irradiance-dependency of PSII quantum yield, as discussed in Silsbe and Kromkamp 2012.

Value

`alpha`	Parameter estimate, standard error, t-value and p-value
`eopt`	Parameter estimate, standard error, t-value and p-value
`ps`	Parameter estimate, standard error, t-value and p-value
`ssr`	Sum of square residuals of fit
`residuals`	Residuals of fit
`model`	EP
`normalize`	Boolean. TRUE or FALSE as passed to the function

Note

Parameter units are dependent on the input.

If normalize=FALSE, then alpha has unit of y/x, eopt has units of x, and ps has units of y.

If normalize=TRUE, then alpha has unit of y, eopt has units of x, and ps has units of y/x.

Author(s)

Greg M. Silsbe

Sairah Y. Malkin

References

Eilers, P.H.C. and Peeters, J.C.H. 1988 A model for the relationship between light intensity and the rate of photosynthesis in phytoplankton. *Ecological Modeling*. **42**, 199–215.

Silsbe, G.M., and Kromkamp, J.C. 2012 Modeling the irradiance dependency of the quantum efficiency of photosynthesis. *Limnology and Oceanography: Methods*. **10**, 642–652.

See Also

fitJP, fitPGH, fitWebb

Examples

```
####   Single PE dataset example    ####

PAR <- c(5,10,20,50,100,150,250,400,800,1200) #umol m-2 s-1
Pc  <- c(1.02,1.99,3.85,9.2,15.45,21.3,28.8,34.5,39.9,38.6) #mg C m-3 hr-1

myfit <- fitEP(PAR, Pc)

#Plot input data
plot(PAR, Pc, xlim=c(0,1500), ylim=c(0,40), xlab="PAR", ylab="Pc")

#Add model fit
E <- seq(0,1500,by=1)
with(myfit,{
  P <- E/((1/(alpha[1]*eopt[1]^2))*E^2+(1/ps[1]-2/(alpha[1]*eopt[1]))*E+(1/alpha[1]))
  lines(E,P)
})


####   Multiple RLC dataset example    ####

data('rlcs')

names(rlcs) #id is unique to a given RLC

id <- unique(rlcs$id)  #Hold unique ids
n  <- length(id)       #5 unique RLCs
```

```
#Setup arrays and vectors to store data

alpha     <- array(NA,c(n,4))
eopt      <- array(NA,c(n,4))
ps        <- array(NA,c(n,4))
ssr       <- rep(NA,n)
residuals <- array(NA,c(n,11))

#Loop through individual RLCs

for (i in 1:n){

  #Get ith data
  PAR  <- rlcs$PAR[rlcs$id==id[i]]
  FqFm <- rlcs$FqFm[rlcs$id==id[i]]

  #Call function
  myfit <- fitEP(PAR,FqFm,normalize=TRUE)

  #Store data
  alpha[i,]     <- myfit$alpha
  eopt[i,]      <- myfit$eopt
  ps[i,]        <- myfit$ps
  ssr[i]        <- myfit$ssr
  residuals[i,] <- myfit$residuals

}
```

---

fitJP *Fit PE or RLC data to Jassby and Platt 1976*

Description

Calculates photosynthetic-irradiance (PE) parameters (alpha, ek) and fit statistics for PE or RLC data using the model of Jassby and Platt 1976

Usage

```
fitJP(x, y, normalize = FALSE, lowerlim = c(0, 1), upperlim = c(100, 1000),
fitmethod=c("Nelder-Mead"))
```

Arguments

x	PAR data. Units of umol m-2 s-1
y	Photosynthetic rate or PSII quantum efficiency.

normalize	Boolean. Default is FALSE. Set to TRUE if fitting PSII quantum efficiency. See Details.
lowerlim	Lower limits of parameter estimates (alpha,ek).
upperlim	Upper limits of parameter estimates (alpha,ek).
fitmethod	The method to be used, one of "Marq", "Port", "Newton", "Nelder-Mead", "BFGS", "CG", "L-BFGS-B", "SANN", "Pseudo". Default is "Nelder-Mead" - see details.

Details

This function passes the data to the function modFIT in the package FME that, through minimization via the specified 'fitmethod' algorithm, determines the optimal model parameters. See the help on modFit algorithms. "Nelder-Mead" is fast and works well for two parameter models, "SANN" is slow and works well for three parameter models.

If normalize is set to FALSE, then data is fit to the equation:

$$y = alpha \times ek \times tanh(\frac{x}{ek})$$

If normalize is set to TRUE, then data is fit to the same equation but normalized to irradiance:

$$y = \frac{1}{x} \times alpha \times ek \times tanh(\frac{x}{ek})$$

Fitting an E-normalized PE model is useful for modeling the irradiance-dependency of PSII quantum yield, as discussed in Silsbe and Kromkamp 2012.

Value

alpha	Parameter estimate, standard error, t-value and p-value
ek	Parameter estimate, standard error, t-value and p-value
ssr	Sum of square residuals of fit
residuals	Residuals of fit
model	JP
normalize	Boolean. TRUE or FALSE as passed to the function

Note

Parameter units are dependent on the input.

If normalize=FALSE, then alpha has unit of y/x and ek has units of x.

If normalize=TRUE, then alpha has unit of y and ek has units of x.

Author(s)

Greg M. Silsbe

Sairah Y. Malkin

References

Jassby, A.D.and Platt, T. 1976 Mathematical formulation of the relationship between photosynthesis and light for phytoplankton. *Limnology and Oceanography*. **21**, 540–547.

Silsbe, G.M., and Kromkamp, J.C. 2012 Modeling the irradiance dependency of the quantum efficiency of photosynthesis. *Limnology and Oceanography: Methods*. **10**, 642–652.

See Also

fitWebb, fitPGH, fitEP

Examples

```
####   Single PE dataset example    ####

PAR <- c(5,10,20,50,100,150,250,400,800,1200) #umol m-2 s-1
Pc  <- c(1.02,1.99,3.85,9.2,15.45,21.3,28.8,34.5,39.9,38.6) #mg C m-3 hr-1

myfit <- fitJP(PAR, Pc)

#Plot input data
plot(PAR, Pc, xlim=c(0,1500), ylim=c(0,40), xlab="PAR", ylab="Pc")

#Add model fit
E <- seq(0,1500,by=1)
with(myfit,{
  P <- alpha[1]*ek[1]*tanh(E/ek[1])
  lines(E,P)
})

####   Multiple RLC dataset example    ####

data('rlcs')

names(rlcs) #id is unique to a given RLC

id <- unique(rlcs$id)  #Hold unique ids
n  <- length(id)       #5 unique RLCs

#Setup arrays and vectors to store data

alpha     <- array(NA,c(n,4))
ek        <- array(NA,c(n,4))
ssr       <- rep(NA,n)
residuals <- array(NA,c(n,11))

#Loop through individual RLCs

for (i in 1:n){

  #Get ith data
```

```
  PAR  <- rlcs$PAR[rlcs$id==id[i]]
  FqFm <- rlcs$FqFm[rlcs$id==id[i]]

  #Call function
  myfit <- fitJP(PAR,FqFm,normalize=TRUE)

  #Store data
  alpha[i,]     <- myfit$alpha
  ek[i,]        <- myfit$ek
  ssr[i]        <- myfit$ssr
  residuals[i,] <- myfit$residuals

}
```

`fitPGH` *Fit PE or RLC data to Platt, Gallegos and Harrison 1980*

Description

Calculates photosynthetic-irradiance (PE) parameters (alpha, beta, ps) and fit statistics for PE or RLC data using the model of Platt, Gallegos and Harrison 1980

Usage

```
fitPGH(x, y, normalize = FALSE, lowerlim = c(0, 0, 0), upperlim = c(100, 1000, 1000),
fitmethod=c("Nelder-Mead"))
```

Arguments

`x`	PAR data. Units of umol m-2 s-1
`y`	Photosynthetic rate or PSII quantum efficiency.
`normalize`	Boolean. Default is FALSE. Set to TRUE if fitting PSII quantum efficiency. See Details.
`lowerlim`	Lower limits of parameter estimates (alpha,beta,ps).
`upperlim`	Upper limits of parameter estimates (alpha,beta,ps).
`fitmethod`	The method to be used, one of "Marq", "Port", "Newton", "Nelder-Mead", "BFGS", "CG", "L-BFGS-B", "SANN", "Pseudo". Default is "Nelder-Mead" - see details.

Details

This function passes the data to the function modFIT in the package FME that, through minimization via the specified 'fitmethod' algorithm, determines the optimal model parameters. See the help on modFit algorithms. "Nelder-Mead" is fast and works well for two parameter models, "SANN" is slow and works well for three parameter models.

If normalize is set to FALSE, then data is fit to the equation:

$$y = ps \times (1 - e^{\frac{-x \times alpha}{ps}}) \times e^{\frac{-x \times beta}{ps}}$$

If normalize is set to TRUE, then data is fit to the same equation but normalized to irradiance:

$$y = \frac{ps}{x} \times (1 - e^{\frac{-x \times alpha}{ps}}) \times e^{\frac{-x \times beta}{ps}}$$

Fitting a E normalized PE model is useful for modeling the irradiance-dependency of PSII quantum yield, as discussed in Silsbe and Kromkamp 2012.

Value

`alpha`	Parameter estimate, standard error, t-value and p-value
`beta`	Parameter estimate, standard error, t-value and p-value
`ps`	Parameter estimate, standard error, t-value and p-value
`ssr`	Sum of square residuals of fit
`residuals`	Residuals of fit
`model`	PGH
`normalize`	Boolean. TRUE or FALSE as passed to the function

Note

Parameter units are dependent on the input.

If normalize=FALSE, then alpha has unit of y/x, beta has units of x, and ps has units of y.

If normalize=TRUE, then alpha has unit of y, beta has units of x, and ps has units of y/x.

Author(s)

Greg M. Silsbe

Sairah Y. Malkin

References

Platt, T., Gallegos, C.L. and Harrison, W.G. 1980 Photoinhibition and photosynthesis in natural assemblages of marine phytoplankton. *Journal of Marine Research*. **38**, 687–701.

Silsbe, G.M., and Kromkamp, J.C. 2012 Modeling the irradiance dependency of the quantum efficiency of photosynthesis. *Limnology and Oceanography: Methods*. **10**, 642–652.

See Also

`fitJP`, `fitWebb`, `fitEP`

Examples

```
####   Single PE dataset example    ####

PAR <- c(5,10,20,50,100,150,250,400,800,1200) #umol m-2 s-1
Pc  <- c(1.02,1.99,3.85,9.2,15.45,21.3,28.8,34.5,39.9,38.6) #mg C m-3 hr-1

#Call function
myfit <- fitPGH(PAR, Pc)

#Plot input data
plot(PAR, Pc, xlim=c(0,1500), ylim=c(0,40), xlab="PAR", ylab="Pc")

#Add model fit
E <- seq(0,1500,by=1)
with(myfit,{
  P <- ps[1]*(1-exp(-1*alpha[1]*E/ps[1]))*exp(-1*beta[1]*E/ps[1])
  lines(E,P)
})


####   Multiple RLC dataset example    ####

data('rlcs')

names(rlcs) #id is unique to a given RLC

id <- unique(rlcs$id)  #Hold unique ids
n  <- length(id)       #5 unique RLCs

#Setup arrays and vectors to store data

alpha     <- array(NA,c(n,4))
beta      <- array(NA,c(n,4))
ps        <- array(NA,c(n,4))
ssr       <- rep(NA,n)
residuals <- array(NA,c(n,11))

#Loop through individual RLCs

for (i in 1:n){

  #Get ith data
  PAR  <- rlcs$PAR[rlcs$id==id[i]]
  FqFm <- rlcs$FqFm[rlcs$id==id[i]]

  #Call function
  myfit <- fitPGH(PAR,FqFm,normalize=TRUE)

  #Store data
  alpha[i,]     <- myfit$alpha
  beta[i,]      <- myfit$beta
```

```
    ps[i,]        <- myfit$ps
    ssr[i]        <- myfit$ssr
    residuals[i,] <- myfit$residuals

  }
```

`fitWebb` *Fit PE data to Webb et al. 1974*

Description

Calculates photosynthetic-irradiance (PE) parameters (alpha, ek) and fit statistics for PE or rapid light curve data using the model of Webb et al. 1974.

Usage

```
fitWebb(x, y, normalize = FALSE, lowerlim = c(0, 1), upperlim = c(100, 1000),
fitmethod=c("Nelder-Mead"))
```

Arguments

`x`	PAR data. Units of umol m-2 s-1
`y`	Photosynthetic rate or PSII quantum efficiency.
`normalize`	Boolean. Default is FALSE. Set to TRUE if y is PSII quantum efficiency. See Details.
`lowerlim`	Lower limits of parameter estimates (alpha,ek).
`upperlim`	Upper limits of parameter estimates (alpha,ek).
`fitmethod`	The method to be used, one of "Marq", "Port", "Newton", "Nelder-Mead", "BFGS", "CG", "L-BFGS-B", "SANN", "Pseudo". Default is "Nelder-Mead" - see details.

Details

This function passes the data to the function modFIT in the package FME that, through minimization via the specified 'fitmethod' algorithm, determines the optimal model parameters. See the help on modFit algorithms. "Nelder-Mead" is fast and works well for two parameter models, "SANN" is slow and works well for three parameter models.

If normalize is set to FALSE, then data is fit to the equation:

$$y = alpha \times ek \times (1 - e^{\frac{-x}{ek}})$$

If normalize is set to TRUE, then data is fit to the same equation but normalized to irradiance:

$$y = \frac{1}{x} \times alpha \times ek \times (1 - e^{\frac{-x}{ek}})$$

Fitting an irradiance-normalized PE model is useful for modeling the irradiance-dependency of PSII quantum yield, as discussed in Silsbe and Kromkamp 2012. If normalize is set to TRUE, x values eqaul to 0 are set to 1e-6.

Value

alpha	Parameter estimate, standard error, t-value and p-value
ek	Parameter estimate, standard error, t-value and p-value
ssr	Sum of square residuals of fit
residuals	Residuals of fit
model	Webb
normalize	Boolean. TRUE or FALSE as passed to the function

Note

Parameter units are dependent on the input.

If normalize=FALSE, then alpha has unit of y/x and ek has units of x.

If normalize=TRUE, then alpha has unit of y and ek has units of x.

Author(s)

Greg M. Silsbe

Sairah Y. Malkin

References

Silsbe, G.M., and Kromkamp, J.C. 2012 Modeling the irradiance dependency of the quantum efficiency of photosynthesis. *Limnology and Oceanography: Methods*. **10**, 642–652.

Webb, W.L., Newton, M., and Starr, D. 1974 Carbon dioxide exchange of *Alnus rubra*: A mathematical model. *Oecologia*. **17**, 281–291.

See Also

fitJP, fitPGH, fitEP

Examples

```
####   Single PE dataset example    ####

PAR <- c(5,10,20,50,100,150,250,400,800,1200) #umol m-2 s-1
Pc  <- c(1.02,1.99,3.85,9.2,15.45,21.3,28.8,34.5,39.9,38.6) #mg C m-3 hr-1

#Call function
myfit <- fitWebb(PAR, Pc)

#Plot input data
plot(PAR, Pc, xlim=c(0,1500), ylim=c(0,40), xlab="PAR", ylab="Pc")

#Add model fit
E <- seq(0,1500,by=1)
with(myfit, {
```

```
  P <- alpha[1] * ek[1] * (1 - exp (-E / ek[1]))
  lines(E,P)
})


####   Multiple RLC dataset example    ####

data('rlcs')

names(rlcs) #id is unique to a given RLC

id <- unique(rlcs$id)  #Hold unique ids
n  <- length(id)       #5 unique RLCs

#Setup arrays and vectors to store data
#All RLCs in example have the same 11 PAR steps in the same order

alpha     <- array(NA,c(n,4))
ek        <- array(NA,c(n,4))
ssr       <- rep(NA,n)
residuals <- array(NA,c(n,11))

#Loop through individual RLCs

for (i in 1:n){

  #Get ith data
  PAR  <- rlcs$PAR[rlcs$id==id[i]]
  FqFm <- rlcs$FqFm[rlcs$id==id[i]]

  #Call function
  myfit <- fitWebb(PAR,FqFm,normalize=TRUE)

  #Store data
  alpha[i,]     <- myfit$alpha
  ek[i,]        <- myfit$ek
  ssr[i]        <- myfit$ssr
  residuals[i,] <- myfit$residuals
}
```

---

incident — *Simulate photosynthetic active radiaton (PAR)*

Description

Derives and simulates PAR over a defined period for a given location.

Usage

```
incident(date, latitude, longitude, elevation, timezone, meanPAR, TL = 3.5,
reflectance = TRUE)
```

Arguments

`date`	The date and time over which to calculate incident PAR. Date and time must be in the ISO format with the timezone set to UTC, see Examples.
`latitude`	Latitude in decimal degrees. Northern hemisphere is positive.
`longitude`	Longitude in decimal degrees. Eastern hemisphere is positive.
`elevation`	Elevation in metres.
`timezone`	Time zone, west is negative. See Details.
`meanPAR`	Optional. Mean daily PAR to scale data to. See Details.
`TL`	Linke turbidity factor that describes atmospheric turbidity. Default value is 3.5, see Details.
`reflectance`	Boolean indicating whether reflectance at the air-water interface should be subtracted from PAR. Default is TRUE, see Details.

Details

Timezone refers to hours relative to UTC. There is currently no provision for daylight savings time. An example for Lake Erie below describes a workaround for daylight savings time.

This function calculates the solar position (azimuth and zenith angle) at each time step using the function insol from the package insolation. Next, shortwave radiation is calculated at each time step as the sum of direct and diffuse radiation following Hofierka and Suri (2002). This calculation requires a Linke turbidity factor (TL) that describes cloud-free atmospheric turbidity. Monthly global maps of TL can be found at http://www.soda-is.com/linke/linke_helioserve.html. Finally, shortwave radiation is multiplied by 2.047 to arrive at PAR (Kirk 2011).

If a meanPAR argument is passed to this function, cloud-free PAR as calculated above is scaled to this value. Mean daily PAR values for a given month can be retrieved from for the global ocean and many large inland lakes from the MODIS ocean colour website http://oceancolor.gsfc.nasa.gov/cgi/l3.

If reflectance=TRUE, then irradiance reflected off the air-water interface is subtracted from PAR, as calculated as a function of zenith angle following Kirk (2011).

Value

A two column matrix specifying the decimal day of year and PAR.

Author(s)

Greg M. Silsbe Sairah Y. Malkin

References

Hofierka, J., and Suri, M. 2002 The solar radiation model for Open source GIS: Implementation and applications. *Proceedings of the source GIS-GRASS users conference 2002.*

Kirk, J.T.O., 2011 Light and photosynthesis in aquatic environments. Cambridge Press.

See Also

reflectance

Examples

```
#Simulate cloud free PAR Lake Diefenbaker on 1 July 2013 using default Linke Turbidity
date <- seq(ISOdatetime(2013,7,1,0,0,0,tz="UTC"),
            ISOdatetime(2013,7,2,0,0,0,tz="UTC"),
            by="10 min")

LD.1 <- incident(date,50,-105,556,-8,reflectance=FALSE)

#Simulate cloud free PAR Lake Diefenbaker on 1 July 2013  Linke Turbidity of 4.5
LD.2 <- incident(date,50,-105,556,-8,TL=4.5,reflectance=FALSE)

#Now simulate PAR for Lake Diefenbaker using a mean PAR value of 578 umol m-2 s-1
LD.3 <- incident(date,50,-105,556,-8,meanPAR=575,reflectance=FALSE)

#Now simulate PAR for Lake Diefenbaker using a mean PAR value of 578 umol m-2 s-1
#and Link Turbidity of 4.5
LD.4 <- incident(date,50,-105,556,-8,meanPAR=575,TL=4.5,reflectance=FALSE)

#Compare simulations
plot(LD.1[,1],LD.1[,2],xlab="Day of year",ylab="PAR",type="l")
lines(LD.2[,1],LD.2[,2],col="red")
lines(LD.3[,1],LD.3[,2],col="blue")
lines(LD.4[,1],LD.4[,2],col="blue",lty=2)

#Simulate annual PAR for Lake Erie, with a workaround for daylight savings time
date1 <- seq(ISOdatetime(2013,1,1,0,0,0,tz="UTC"),
             ISOdatetime(2013,3,9,0,0,0,tz="UTC"),
             by="30 min")

date2 <- seq(ISOdatetime(2013,3,9,0,0,0,tz="UTC"),
             ISOdatetime(2013,11,2,0,0,0,tz="UTC"),
             by="30 min")

date3 <- seq(ISOdatetime(2013,11,2,0,0,0,tz="UTC"),
             ISOdatetime(2014,1,1,0,0,0,tz="UTC"),
             by="30 min")

LE <- rbind(incident(date1,42.15,-81,115,-5,reflectance=FALSE),
            incident(date2,42.15,-81,115,-4,reflectance=FALSE),
            incident(date3,42.15,-81,115,-5,reflectance=FALSE))
```

```
#plot data
plot(LE[,1],LE[,2],xlab="Day of year",ylab="PAR",type="l")
```

`phytoprod` *Calculates phytoplantkon production.*

Description

Calculates phytoplankton production as a function of incident irradiance, an attenuation coefficient (kpar), photosynthetic-irradiance (PE) parameters, and an optional biomass profile.

Usage

```
phytoprod(PE, Ein, kpar, cz = matrix( data=c(1,1), ncol = 2), zmax = NA)
```

Arguments

`PE`	A list returned by either `fitEP`, `fitJP`, `fitPGH`, or `fitWebb`.
`Ein`	A two column matrix specifying the decimal day of year and PAR. The same format as returned by `incident`.
`kpar`	The attenuation coefficient of PAR. Units are m-1.
`cz`	Optional. A two column matrix specifying depth in column 1 and biomass in column 2. See Examples.
`zmax`	Optional. The maximum depth of integration. See Details.

Details

Units are dependent on the PE input.

If a zmax value is passed to the function and is shallower than the computed euphotic depth (defined here as 0.5 If a zmax value is not passed to the function or the specified value is deeper than the computed euphotic depth, then vertical integration is constrained to the euphotic depth.

If PE has noramalize=FALSE, then P has units of x m-3 hr-1 and PP has units of x m-2 day-1, where x is the original units of P passed to the fitPE function.

If PE has noramalize=TRUE, then P has units of mmol photons m-3 hr-1 and PP has units of mmol photons m-2 day-1.

Value

`PP`	A matrix specifying day of year and areal phytoplankton production
`z`	A vector specifying the depths over which photosynthetic rates are calculated
`t`	A vector specifying the times over which photosynthetic rates are calculated
`P`	A matrix of dimension [t,z] containing photosynthetic rates

Author(s)

Greg M. Silsbe

Sairah Y. Malkin

See Also

incident, fitWebb, fitJP, fitPGH, fitEP

Examples

```
#Model incident irradiance for Lake Superior on July 31, 2007
date <- seq(ISOdatetime(2013,7,31,0,0,0,tz="UTC"),
            ISOdatetime(2013,8,1,0,0,0,tz="UTC"),
            by="10 min")

E0 <- incident(date, 47.33, -89.8, 180, -6, meanPAR=480, reflectance=TRUE)

plot(E0[,1],E0[,2],type="l")

#Model PE data
P <- c(0.64,1.32,1.09,0.53,0.37,0.17,0.02)/24   #(umol C ug chla-1 hr-1)
E <- c(373,255,136,38.6,10.95,3.1,0.25)         #(umol m-2 s-1)

#Fit data to Eilers and Peeters
myfit1 <- fitEP(E,P)

#Fit data to Jasbby and Platt
myfit2 <- fitJP(E,P)

#Plot PE curve
plot(E,P)
E <- c(0:400)
#Eilers and Peeters
P.EP <- E/((1/(myfit1$alpha[1]*myfit1$eopt[1]^2))*E^2+
           (1/myfit1$ps[1]-2/(myfit1$alpha[1]*myfit1$eopt[1]))*E+
           (1/myfit1$alpha[1]))
lines(E,P.EP,col="red")
#Jassby and Platt
P.JP <- myfit2$alpha[1]*myfit2$ek[1]*tanh(E/myfit2$ek[1])
lines(E,P.JP,col="blue")

#Compare Areal Primary production between two fits
#Assume constant chlorophyll through depth of 0.894 ug/L

#Eilers and Peeters
phytoprod(myfit1,
          E0,
          kpar=0.126,
          cz=matrix(data=c(1,0.894),ncol=2))$PP
#Units of umol C m-2 day-1
```

```
#Jassby and Platt
phytoprod(myfit2,
          E0,
          kpar=0.126,
          cz=matrix(data=c(1,0.894),ncol=2))$PP
#Units of umol C m-2 day-1

#Now let chlorophyll change with depth
cz <- matrix(data=c(0.462,0.699,1.065,1.332,1.245,1.156,0.636,0.558,
                    2,5,10,20,30,40,60,80),ncol=2)

myPP <- phytoprod(myfit1,
                  E0,
                  kpar=0.126,
                  cz,
                  zmax=80)

myPP$PP #Units of umol C m-2 day-1

#Plot photosynthetic rate through depth
#Units of umol C m-3 hr-1

image(x=myPP$t,
      y=myPP$z,
      z=myPP$P,
      col=rev(heat.colors(20)),
      ylim=c(80,0),
      zlim=c(1e-5,0.1),
      xlab="Decimal Day",
      ylab="Depth (m)")
```

---

reflectance *Computes surface reflectance*

Description

The fraction of surface reflectance at the air-water interface is calculated as a function of solar zenith angle following Kirk (2011).

Usage

```
reflectance(date, latitude, longitude, timezone)
```

Arguments

`date`	The date and time over which to calculate reflectance. Date and time must be in the ISO format with the timezone set to UTC, see examples.
`latitude`	Latitude in decimal degrees. Northern hemisphere is positive.

`longitude`	Longitude in decimal degrees. Eastern hemisphere is positive.
`timezone`	Time zone, west is negative. See Details.

Details

Timezone refers to hours relative to UTC. There is currently no provision for daylight savings time.

Value

A two column matrix specifying the decimal day of year and the fraction of surface reflectance.

Author(s)

Greg M. Silsbe Sairah Y. Malkin

References

Kirk, J.T.O., 2011. Light and photosynthesis in aquatic environments. Cambridge Press.

See Also

incident

Examples

```
#Calculate surface reflectance in 10 minute increments
#for Godthabsfjord fjord, Greenland on March 1

#Setup date sequence
date <- seq(ISOdatetime(2013,3,1,0,0,0,tz="UTC"),
            ISOdatetime(2013,3,2,0,0,0,tz="UTC"),
            by = "10 min")

#Call the function
ref <- reflectance(date,64.20,-51.76,-3)

#Plot data
plot(ref[,1],ref[,2],type="l",xlab="Day of Year",ylab="Surface Reflectance")
```

---

rlcs   *Sample rapid light curves*

Description

A sample dataset containing 5 rapid light curves.

Usage

```
data(rlcs)
```

Format

A dataframe with 3 columns, id, PAR and FqFm.

Index