

photobiologyWavebands Version 0.4.0

User Guide

Pedro J. Aphalo

February 2, 2016

1 Introduction

This package provides constructors for **waveband** objects, a class defined in package **photobiology**. These are convenience functions that allow definition of range of wavelengths and biological spectral weighting functions (BSWFs) following definitions in common use and ISO standards.

2 Calculating irradiances

Functions for several colour bands, in some cases according to different optional definitions, are listed in Table 1.

An example using **sun.spct** included in package **photobiology**. As the input spectral irradiance is units of $\text{W m}^{-2} \text{nm}^{-1}$ the output is in $\text{mol m}^{-2} \text{s}^{-1}$ or W m^{-2} .

```
e_irrad(sun.spct, UV()) # W m-2

## UV.ISO.tr.lo
##      28.62872
## attr(,"time.unit")
## [1] "second"
## attr(,"radiation.unit")
## [1] "energy irradiance total"

q_irrad(sun.spct, UV()) * 1e6 # umol s-1 m-2

## UV.ISO.tr.lo
##      86.49506
## attr(,"time.unit")
## [1] "second"
## attr(,"radiation.unit")
## [1] "photon irradiance total"
```

Irradiances for different wavebands can be grouped into a list of any length. If the list has named members, then these names are used instead of the default ones.

Table 1: Functions in R package `photobiologyWavebands` used for constructing descriptors of wavebands used for calculation of irradiances or exposures. The boundaries of the band given as wavelengths in nm (λ). Definition according to ISO-21348 "ISO" is the default for all functions except `PAR()` for which there is only one definition in common use "Plant" which is the default.

Waveband	Source	Dose or irradi.	Waveband (nm)
UV	ISO-21348	UV("ISO")	$100 \leq \lambda < 400$
UV-C	ISO-21348	UVC("ISO")	$100 \leq \lambda < 280$
UV-C	n.a.	UVC("medical")	$220 \leq \lambda < 290$
UV-C	n.a.	UVC("none")	$200 \leq \lambda < 280$
UV-B	ISO-21348	UVB("ISO")	$280 \leq \lambda < 315$
UV-B	n.a.	UVB("none")	$280 \leq \lambda < 320$
UV-A	ISO-21348	UVA("ISO")	$315 \leq \lambda < 400$
UV-A	n.a.	UVA("none")	$320 \leq \lambda < 400$
Visible	ISO-21348	VIS("ISO")	$380 \leq \lambda < 760$
Photosynthesis	n.a.	PAR("Plant")	$400 \leq \lambda < 700$
Purple	ISO-21348	Purple("ISO")	$360 \leq \lambda < 450$
Blue	Sellaro	Blue("Sellaro")	$420 \leq \lambda < 490$
Blue	ISO-21348	Blue("ISO")	$450 \leq \lambda < 500$
Green	Sellaro	Green("Sellaro")	$500 \leq \lambda < 570$
Green	ISO-21348	Green("ISO")	$500 \leq \lambda < 570$
Yellow	ISO-21348	Yellow("ISO")	$570 \leq \lambda < 591$
Orange	ISO-21348	Orange("ISO")	$591 \leq \lambda < 610$
Red	ISO-21348	Red("ISO")	$610 \leq \lambda < 760$
Red	Smith	Red("Smith10")	$655 \leq \lambda < 665$
Red	Smith ?	Red("Smith20")	$650 \leq \lambda < 670$
Red	Inada	Red("Inada")	$600 \leq \lambda < 700$
Red	Warrington	Red("Warrington")	$625 \leq \lambda < 675$
Red	Sellaro	Red("Sellaro")	$620 \leq \lambda < 680$
Far-red	ISO	Far_red("ISO")	not defined
Far-red	Smith10	Far_red("Smith")	$725 \leq \lambda < 735$
Far-red	Smith20	Far_red("Smith")	$720 \leq \lambda < 740$
Far-Red	Inada	Red("Inada")	$700 \leq \lambda < 800$
Far-Red	Warrington	Red("Warrington")	$700 \leq \lambda < 850$
Far-red	Sellaro	Far_red("Sellaro")	$700 \leq \lambda < 750$
Far-red	BTV	Far_red("BTV")	$700 \leq \lambda < 760$
<i>Arbitrary</i>	n.a.	<code>new_waveband(lo,hi)</code>	$lo \leq \lambda < hi$

Table 2: Functions in R package `photobiologyWavebands` used for constructing lists descriptors of wavebands used for calculation of irradiances or exposures.

Waveband	Source	Function	std values used
VIS defs.	ISO-21348	<code>VIS_bands("ISO")</code>	"ISO"
UV defs.	ISO-21348	<code>UV_bands("ISO")</code>	"ISO"
UV defs.	n.a.	<code>UV_bands("none")</code>	"none"
Plant sens.	n.a.	<code>Plant_bands("sensory20")</code>	"ISO", "Sellaro" and "Smith20"
Plant sens.	n.a.	<code>Plant_bands("sensory")</code>	"ISO", "Sellaro" and "Smith20"
Plant sens.	n.a.	<code>Plant_bands("sensory10")</code>	"ISO", "Sellaro" and "Smith10"
Plant sens.	n.a.	<code>Plant_bands("energy")</code>	"ISO" and "McCree"

```
e_irrad(sun.spct, list(Blue(), VIS()))

## Blue.ISO   VIS.ISO
## 37.55207 231.86345
## attr("time.unit")
## [1] "second"
## attr("radiation.unit")
## [1] "energy irradiance total"

e_irrad(sun.spct, list(B = Blue(), VIS()))

## Blue.ISO   VIS.ISO
## 37.55207 231.86345
## attr("time.unit")
## [1] "second"
## attr("radiation.unit")
## [1] "energy irradiance total"
```

A few functions for generating coherent lists of wavebands are also defined (Table 2).

```
e_irrad(sun.spct, VIS_bands())

## Purple.ISO   Blue.ISO   Green.ISO   Yellow.ISO
## 47.75529    37.55207    49.26860    13.67971
## Orange.ISO   Red.ISO
## 12.00432     79.38159
## attr("time.unit")
## [1] "second"
## attr("radiation.unit")
## [1] "energy irradiance total"
```

3 Photon ratios

Photon ratios can be calculated from any pair of waveband objects. This a convenient and very flexible way of doing this type of calculations.

```
q_ratio(sun.spct, Blue(), VIS())

## Blue.IS0: VIS.IS0(q:q)
##           0.1371157
## attr(,"radiation.unit")
## [1] "q:q ratio"
```

4 Effective irradiances and exposures

The waveband definitions and SWFs are stored in **waveband** objects, that can be created with function **waveband**. The same functions described above for unweighted irradiances are used to calculate effective irradiances and doses.

Currently functions for constructing **waveband** objects describing several BSWFs are implemented (see Table 3). These functions take three arguments in most cases as they have been used and continue to be used inconsistently in the scientific literature. By supplying these arguments different variations of the BSWFs can be obtained. The defaults used are those values which we consider best, usually the most frequently used ones, except in cases when we consider the use of those values problematic for the reliability of the calculations.

```
e_irrad(sun.spct, CIE())

## CIE98.298.tr.lo
##           0.08181583
## attr(,"time.unit")
## [1] "second"
## attr(,"radiation.unit")
## [1] "energy irradiance total"
```

5 Action spectra at given wavelengths

The functions available for calculating action spectra take as argument a vector of wavelengths, and return a vector of effectiveness (either quantum/photon or energy based) depending on how the original source describes them. These functions are listed in Table 4, and an example of their use follows.

```
# at 1 nm intervals
wavelengths1 <- 285:400
action.spectrum1 <- CIE_e_fun(wavelengths1)
```

All functions accept a wavelengths vector with variable and arbitrary step sizes, with the condition that the wavelengths are sorted in strictly increasing order.

In practice these functions are mainly used internally by the package, and very rarely in user code, as the same output can be obtained by multiplication of **source_spct** objects by **waveband** objects.

Compare the following two plots,

Table 3: Functions in R package `photobiologyWavebands` used for constructing `waveband` objects describing BSWFs used for calculation of effective irradiances or doses. The functions for BSWFs available in this package, are by default as in the original source. Optionally they can be normalized to any wavelength within their non-zero range by providing the `norm` argument with a wavelength in nm. The range of wavelengths included when calculating integrals is given by `w.low` and `w.high`. The values in the table below are the defaults.

Action spectrum	Formulation	Constructor function	norm λ (nm)	w.low λ (nm)	w.high λ (nm)
Gen. plant action	Green	GEN_G(norm, w.low, w.high)	300	275	313.3
Gen. plant action	Thimijan	GEN_T(norm, w.low, w.high)	300	275	345.0
Gen. plant action	Micheletti	GEN_M(norm, w.low, w.high)	300	275	313.3
Plant growth	Flint & Caldwell	PG(norm, w.low, w.high)	300	275	390.0
Erythema	CIE98	CIE(norm, w.low, w.high)	298	250	400.0
ICNIRP	ICNIRP2004	ICNIRP(norm, w.low, w.high)	270	210	400.0
'Naked' DNA	TUV, from Setlow	DNA_N(norm, w.low, w.high)	300	250	400.0
'Naked' DNA	Green & Miller	DNA_GM(norm, w.low, w.high)	300	250	400.0
'Plant' DNA	Musil, from Quate	DNA_P(norm, w.low, w.high)	300	250	400.0
Flavonoid	Ibdah	FLAV(norm, w.low, w.high)	300	275	346.0

Table 4: Biological spectral weighting functions predefined in R package **photobiologyWavebands**. The functions for BSWFs available in this package, implement the functions as defined in the original publications. When the original ‘definition’ is available as tabulated data, or we have tabulated it by digitizing a figure, the values returned are calculated by spline interpolation.

Action spectrum	Formulation	Function	Norm. λ (nm)
Gen. plant action	Green	GEN_G_q_fun(w.length)	280
Gen. plant action	Thimijan	GEN_T_q_fun(w.length)	300
Gen. plant action	Micheletti	GEN_M_q_fun(w.length)	300
Plant growth	Flint & Caldwell	PG_q_fun(w.length)	300
Erythema	CIE98	CIE_e_fun(w.length)	298
ICNIRP	ICNIRP2004	ICNIRP_e_fun(w.length)	270
‘Naked’ DNA	TUV, from Setlow	DNA_N_q_fun(w.length)	n.a.
‘Naked’ DNA	Green & Miller	DNA_GM_q_fun(w.length)	n.a.
‘Plant’ DNA	Musil, from Quate	DNA_P_q_fun(w.length)	290
Flavonoid	Ibdah	FLAV_q_fun(w.length)	300

```
sun.spct * CIE()

## Object: source_spct [122 x 2]
## Wavelength (nm): range 280 to 400, step 0.9230769 to 1
## Time unit: 1s
## Data weighted using 'CIE98.298' BSWF
##
##   w.length s.e.irrad
##   (dbl)    (dbl)
## 1 280.0000      0
## 2 280.9231      0
## 3 281.8462      0
## 4 282.7692      0
## 5 283.6923      0
## 6 284.6154      0
## 7 285.5385      0
## 8 286.4615      0
## 9 287.3846      0
## 10 288.3077      0
## ..      ...      ...
```

6 Luminous flux

The luminous flux per unit area in lux can be calculated as follows using the original luminous efficiency function for the human eye using for defining the lumen. As we start with spectral irradiance we obtain luminous flux per unit area expressed in lux.

```
e_response(sun.spct * CIE1924_lef.spct) * photopic_sensitivity

##      Total
## 49579.93
## attr(,"time.unit")
## [1] "second"
## attr(,"radiation.unit")
## [1] "energy response total"
```

The luminous flux per unit area in lux can be calculated as follows using the latest luminous efficiency function for the human eye.

```
e_response(sun.spct * CIE2008_lef2deg.spct) * photopic_sensitivity

##      Total
## 53057.78
## attr(,"time.unit")
## [1] "second"
## attr(,"radiation.unit")
## [1] "energy response total"
```

As the luminous efficiency functions vary slightly in the wavelength at which the maximum is located, and the wavelength used for the sensitivity constant is fixed by the definition of the Lumen, a small correction is need for exact results.

```
e_response(sun.spct * CIE2008_lef2deg.spct) * photopic_sensitivity *
  interpolate_spct(CIE2008_lef2deg.spct, 555)$s.e.response

##      Total
## 53910.01
## attr(,"time.unit")
## [1] "second"
## attr(,"radiation.unit")
## [1] "energy response total"
```

An equivalent quantity can be calculated for scotopic vision, using the corresponding function and constant.

```
e_response(sun.spct * 1e-6 * CIE1951_scotopic_lef.spct) * scotopic_sensitivity

##      Total
## 0.1186256
## attr(,"time.unit")
## [1] "second"
## attr(,"radiation.unit")
## [1] "energy response total"
```