

# photobiologyFilters Version 0.2.1

## User Guide

Pedro J. Aphalo

January 23, 2015

## 1 Introduction

We have developed a set of packages to facilitate the calculation of many different quantities that can be derived from spectral irradiance data. The basic package is called **photobiology**, and the package described here adds transmittance data for some frequently used filters, and a function for interpolating.

## 2 Installation and use

The functions in the package **photobiologyFilters** are made available by installing the packages **photobiologyFilter** (once) and loading it from the library when needed.

To load the package into the workspace we use `library(photobiologyFilter)`.

```
library(photobiology)
library(photobiologygg)
library(photobiologyFilters)
library(ggplot2)
```

## 3 Calculating average transmittance

```
transmittance(uv.226.new.spct, UV_bands())

##      UVB.ISO      UVA.ISO
## 0.00000000 0.04767941
## attr("time.unit")
## [1] "none"
## attr("Tfr.type")
## [1] "total"
## attr("radiation.unit")
## [1] "transmittance average"
```

```
transmittance(uv.226.new.spct, PAR())
```

```
##      PAR
## 0.8819025
## attr("time.unit")
## [1] "none"
## attr("Tfr.type")
## [1] "total"
## attr("radiation.unit")
## [1] "transmittance average"
```

```
absorbance(uv.226.new.spct, PAR())
```

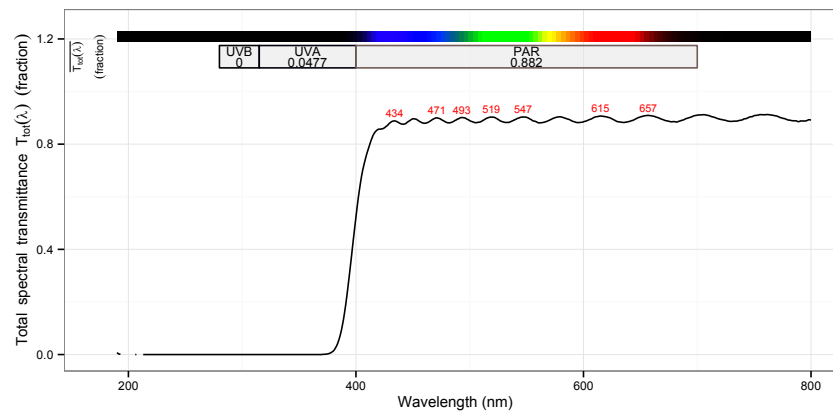
```
## Warning: NaNs produced
```

```
##      PAR
## 0.05526775
## attr("time.unit")
## [1] "none"
## attr("Tfr.type")
## [1] "total"
## attr("radiation.unit")
## [1] "absorbance average"
```

## 4 Plotting the transmittance spectrum of filters

We can plot the data as is, accepting all defaults. As the returned value is a ggplot object, we can modify and add to it as needed.

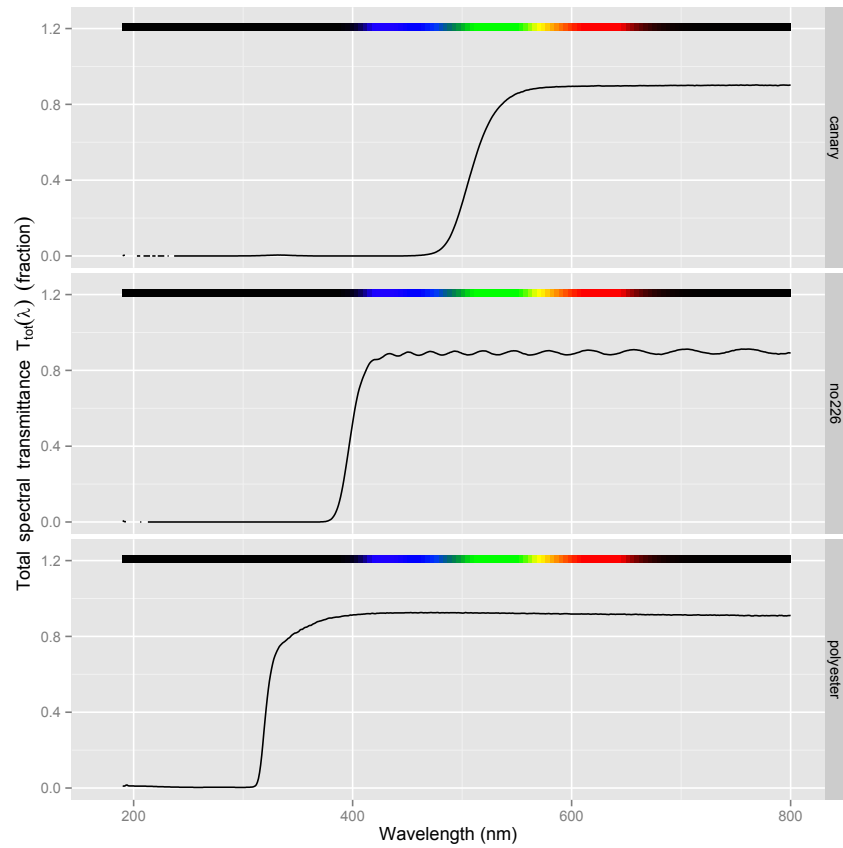
```
plot(uv.226.new.spct) + theme_bw(10)
```



We can also row-bind several spectra and plot them in panels.

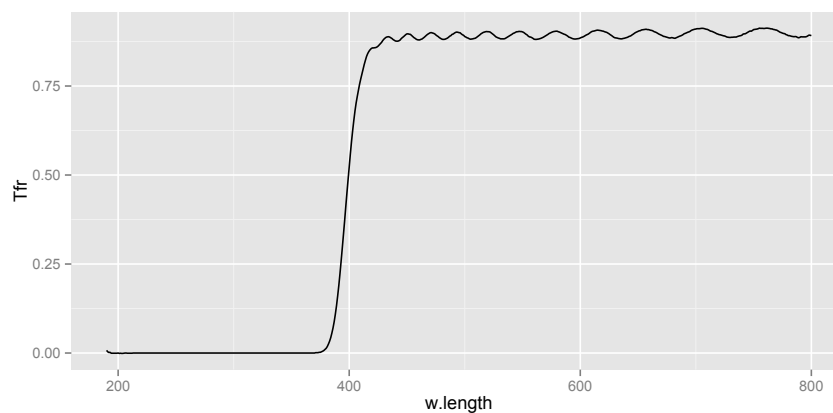
```
filters.spct <-
  rbindspct(list(canary = canary.yellow.new.spct,
                 polyester = polyester.new.spct,
                 no226 = uv.226.new.spct),
            add.factor = TRUE,
            factor.name = "filter")
```

```
plot(filters.spct, annotations = "colour_guide") + facet_grid(filter~.)
```



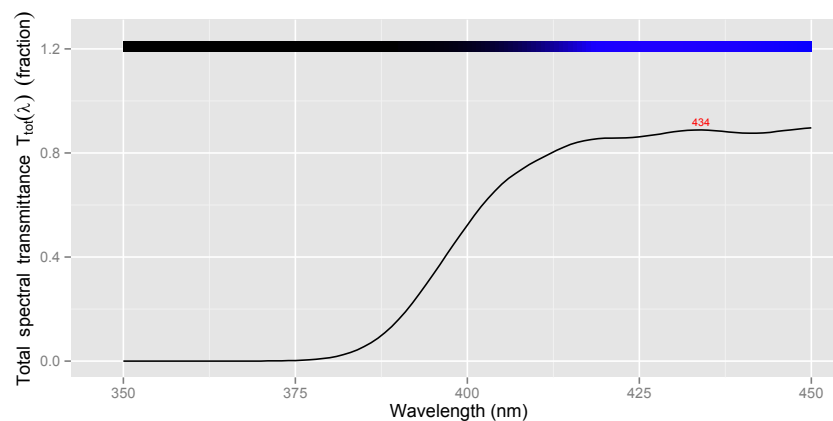
We can also use ggplot (or lattice or base graphics functions) to plot the spectrum.

```
ggplot(data=uv.226.new.spct, aes(x=w.length, y=Tfr)) + geom_line()
```



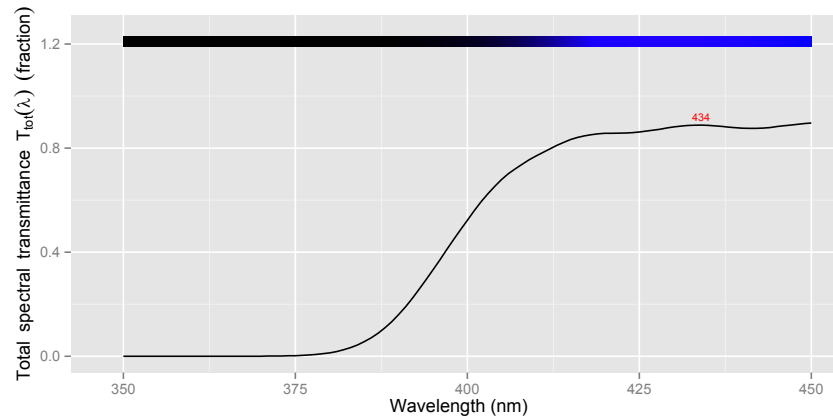
If we would like to use a different range of wavelengths for several calculations, then we can build a new filter.spct by trimming.

```
cp.uv.226.spct <- copy(uv.226.new.spct)
my.uv.226.spct <- trim_spct(cp.uv.226.spct, range = c(350,450))
plot(my.uv.226.spct)
```



However, if we only would like to limit the range of wavelengths in the plot, it is even easier to use the **range** formal argument of function **plot** to obtain the same plot.

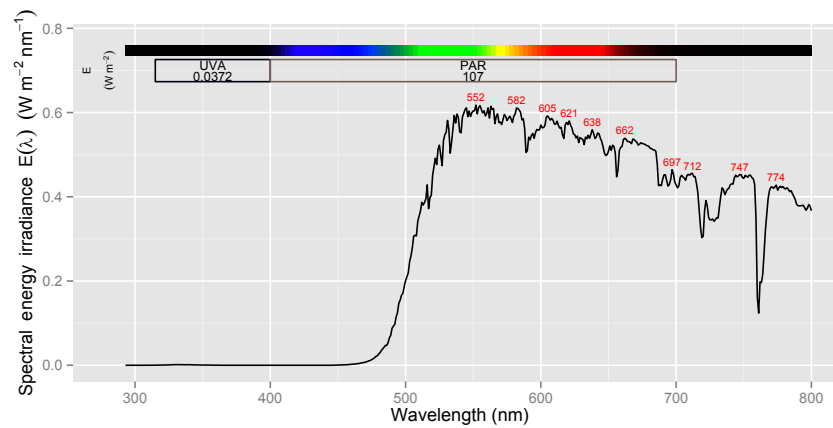
```
plot(uv.226.new.spct, range = c(350, 450))
```



## 5 Convoluting spectral irradiance with filter transmittance

To simulate the spectral irradiance of a filtered light source we convolute the bare source's irradiance spectrum with the filter transmittance spectrum.

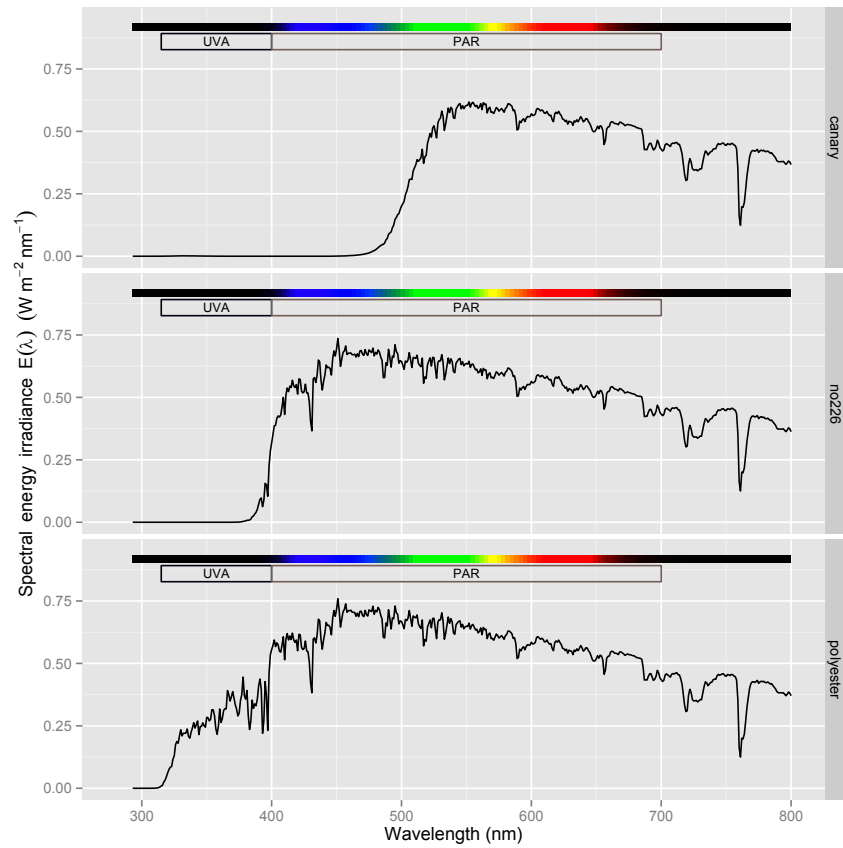
```
plot(sun.spct * canary.yellow.new.spct)
```



If we want to compare the effect of several filters on the spectrum of a light source we can create a combined spectral object by row-binding the source.spct objects obtained as a result of the convolutions.

```
all.spct <-
  rbindspct(list(canary = sun.spct * canary.yellow.new.spct,
    polyester = sun.spct * polyester.new.spct,
    no226 = sun.spct * uv.226.new.spct),
    add.factor = TRUE,
    factor.name = "filter")

plot(all.spct, annotations = c("colourguide", "boxes", "labels")) + facet_grid(filter~.)
```



If we would like the different spectra plotted on a single panel, we can use, for example, package `ggplot2`.

```
ggplot(data=all.spct, aes(x=w.length, y=s.e.irrad, colour=filter)) + geom_line()
```

