P lividus larvae light levels

John Kirwan

3/19/2021

Need to determine the appropriate light level for sea urchin larvae during the night and day phase of their diel cycle. According to Maurizio, the larvae are probably found at 100 m depth during the day and close to the surface at night. Need to calculate the photon irradiance in the appropriate wavelength range and provide similar light to the animals using aquarium lights.

Yet, the gulf of Naples is not 100 m deep - where are larvae likely to reside during the day? Moreover, if light is the sole guide - the light level of water will vary dramatically.

LTER-MC: Marechiara sampling point. > The sampling site LTER-MC is located 2 miles offshore the city of Naples, on the 75 m isobath, at the border between the coastal eutrophic system influenced by land runoff and the offshore oligotrophic waters with characteristics of the southern Tyrrhenian Sea.

Climatological irradiance data from Gulf of Naples The following data come from Hydrolight, courtesy of Maurizio Ribera.

> Units are microEinstein m^-2 s^-1. E_d is downwelling (cosine) E0_ is scalar, both in absolute value at given depth. R and F are for Raman scattering a Fluorescence 0 not accounted for 1 accounted for (I am sending those with 1,1). The incident irradiance is the cloud free irradiance for that date at the site.

The values are **per nm in the binwidth range**

> The incident irradiance is the cloud free irradiance for that date at the site. However E_d(0) is the E_d(0^-). Since the profiles are all at noon you can assume that E_d(0^-) =0.98 E_d(0^+) the incident irradiance at sea surface.

The directed irradiance given as 0m is actually below the surface and should be 98% of the value just above the surface.

The values include both Raman scattering and fluorescence. The incident irradiance is the cloud free irradiance for that date at the site at noon.

Ed is *directed irradiance.* E0 is *scalar irradiance.*

> The technology of measurement of scalar irradiance is essentially the same as that of irradiance with the exception of the collector. Since this must respond equally to light from all directions, it has to be spherical.
>
> Once you have absolutely calibrated your system, irradiance measurements are straightforward. Remember from chapter 2 that two kinds of irradiance are the most useful, vector and scalar. Vector irradiance measures light hitting a surface from all angles in one hemisphere, but weights the amount of the light from each direction by the cosine of its angle from the perpendicular to the surface. Scalar irradiance measures all the light intersecting a surface from the whole sphere, giving all directions equal weight. Both properties are measured using special sampling optics.

> Scalar irradiance does not suffer from the orientation dependence of vector irradiance. Scalar irradiance is given by the number of photons that pass through the surface of a small sphere over a given period of time. It has the same units as vector irradiance, but the directions of the photons do not matter. So, for any location in space there is only one scalar irradiance, while vector irradiance depends on the orientation of the detector.

For these purposes, scalar irradiance makes more sense.

Those who study phytoplankton use *scalar irradiance*. In studies of photosynthesis, one *Einstein* is a mole of photons.

The wavelength bins are 5 nm wide and the stated value is the midpoint of the bin, i.e. the minimum value of 352.5 refers to the 350 - 355 nm bin, whereas the maximum value 862.5 refers to the bin from 860 - 865 nm. The values are nonetheless per nm.

We bring in the data. I had interpolated the irradiance at 100 m by taking the decrement from 50 m to 75 m at each wv bin (25-30%) and applying this to the 25 m below.

> It is definitely reasonable to obtain deeper value using the average $K(\lambda)$ over, say, 50 to 60 m.
>
> – Maurizio of Montesanto

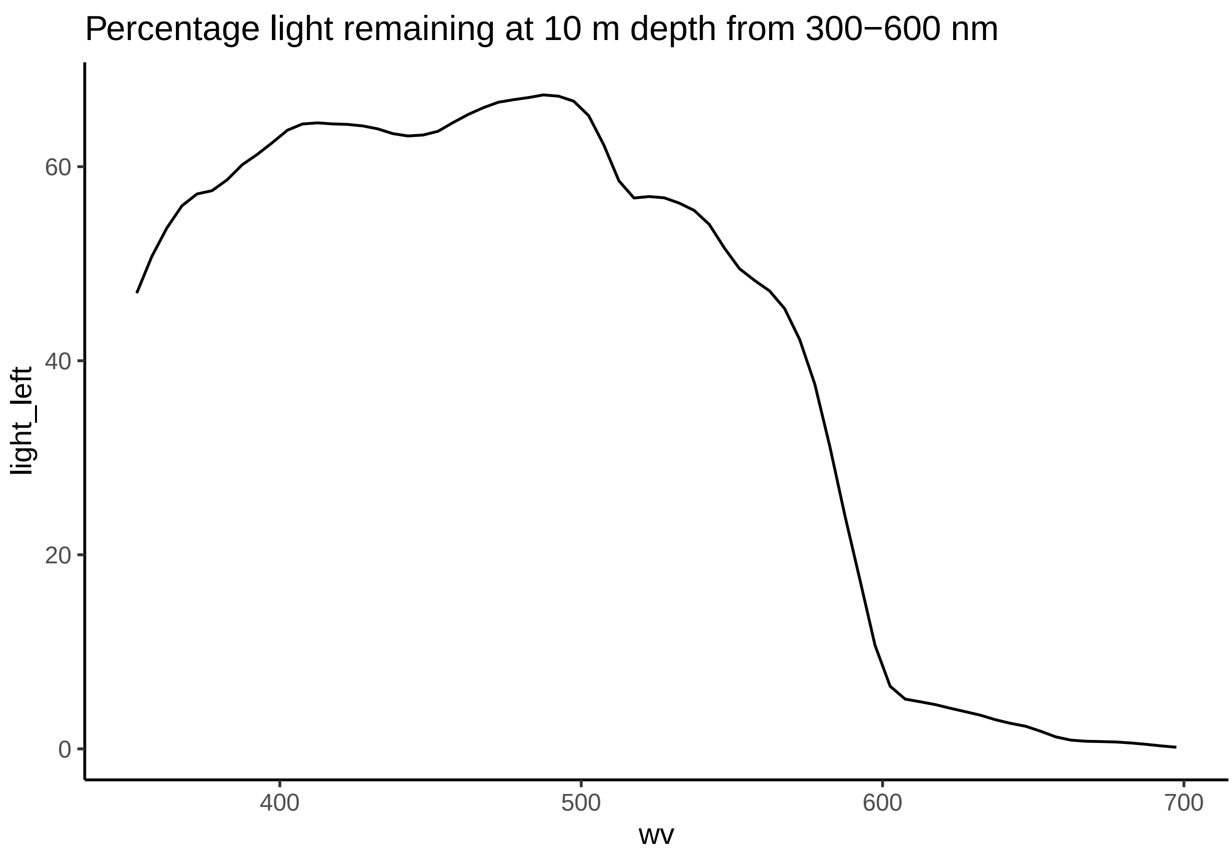
Therefore, the interpolation is over the preceding 50 m, the proportion of light remaining being squared to get the change over the subsequent 25 m - the two results being very similar.

```
#seq1 <- seq(from=0,to=75,by=5)
E0_df <- readr::read_delim('MC_E0_R1F1.txt',delim=',')#,# col_names = c('wv',seq1),skip=1)
colnames(E0_df)[1] <- 'wv'
#E0_df$`100` <- ( E0_df$`75` / E0_df$`50` ) * E0_df$`75`  # 25 m change at depth
E0_df$`100` <- sqrt( E0_df$`75`/E0_df$`25`) * E0_df$`75` # 25 m change at depth
E0_df$`125` <- ( E0_df$`75` / E0_df$`50` ) * E0_df$`100` # 25 m change at depth
E0_df$`150` <- ( E0_df$`75` / E0_df$`50` ) * E0_df$`125` # 25 m change at depth
head(E0_df)
```

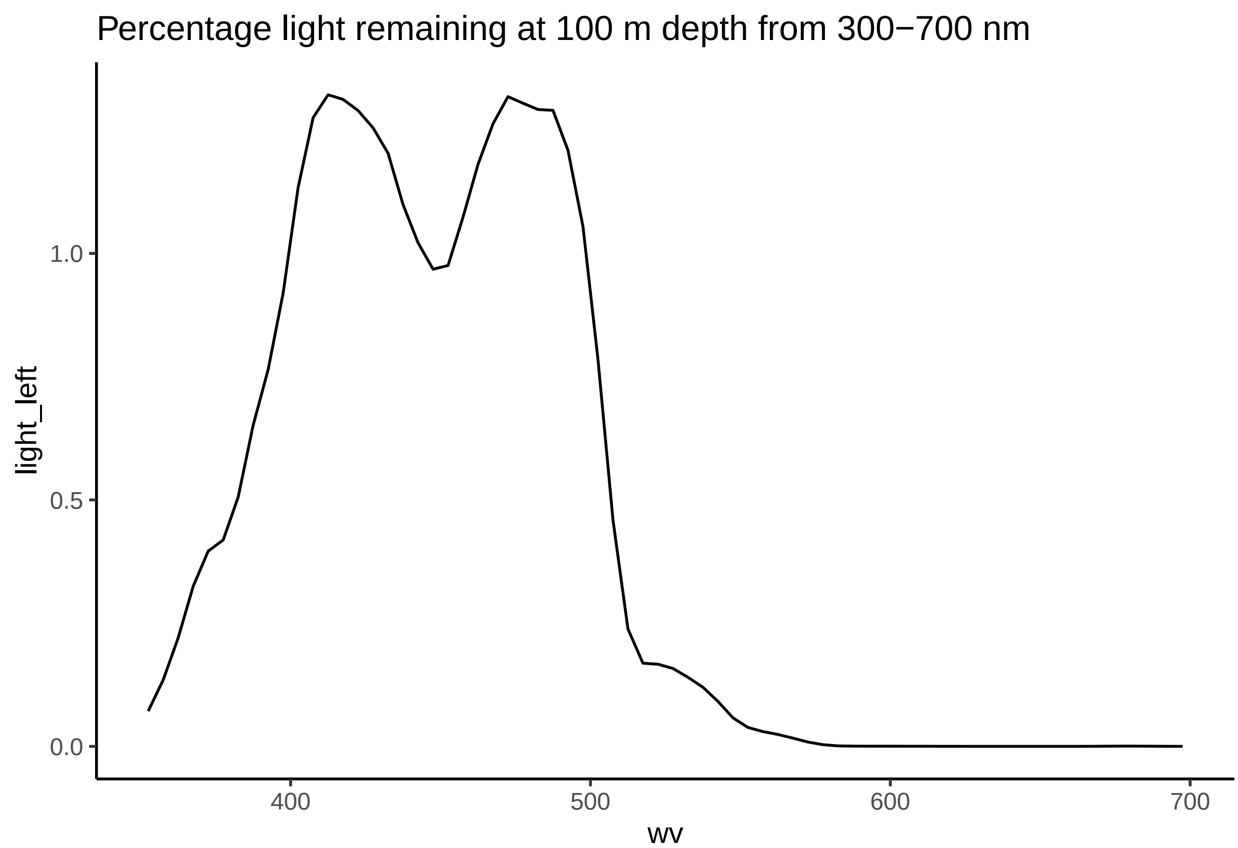
```
## # A tibble: 6 x 20
##      wv   `0`   `5`  `10`  `15`  `20`  `25`   `30`   `35`   `40`   `45`   `50`
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
## 1  352. 0.808 0.568 0.380 0.250 0.165 0.110 0.0746 0.0516 0.0364 0.0259 0.0184
## 2  358. 0.721 0.526 0.366 0.250 0.171 0.119 0.0832 0.0594 0.0432 0.0317 0.0232
## 3  362. 0.811 0.608 0.435 0.307 0.216 0.154 0.111  0.0809 0.0602 0.0452 0.0338
## 4  368. 0.956 0.732 0.535 0.385 0.277 0.201 0.148  0.110  0.0838 0.0640 0.0489
## 5  372. 0.827 0.639 0.473 0.344 0.251 0.184 0.137  0.103  0.0789 0.0609 0.0469
## 6  378. 0.919 0.712 0.529 0.386 0.282 0.207 0.154  0.117  0.0897 0.0694 0.0536
## # ... with 8 more variables: `55` <dbl>, `60` <dbl>, `65` <dbl>, `70` <dbl>,
## #   `75` <dbl>, `100` <dbl>, `125` <dbl>, `150` <dbl>
```

What percentage of light remains compared to the surface?

```
E0_df %>%
  filter(between(wv,300,700)) %>%
  mutate(light_left = 100*`10` / `0`) %>%
  ggplot(aes(y=light_left,x=wv)) + geom_line() +
  ggtitle('Percentage light remaining at 10 m depth from 300-600 nm') + theme_classic()
```




```
E0_df %>%
  filter(between(wv,300,700)) %>%
  mutate(light_left = 100*`100` / `0`) %>%
  ggplot(aes(y=light_left,x=wv)) + geom_line() +
  ggtitle('Percentage light remaining at 100 m depth from 300-700 nm') + theme_classic()
```



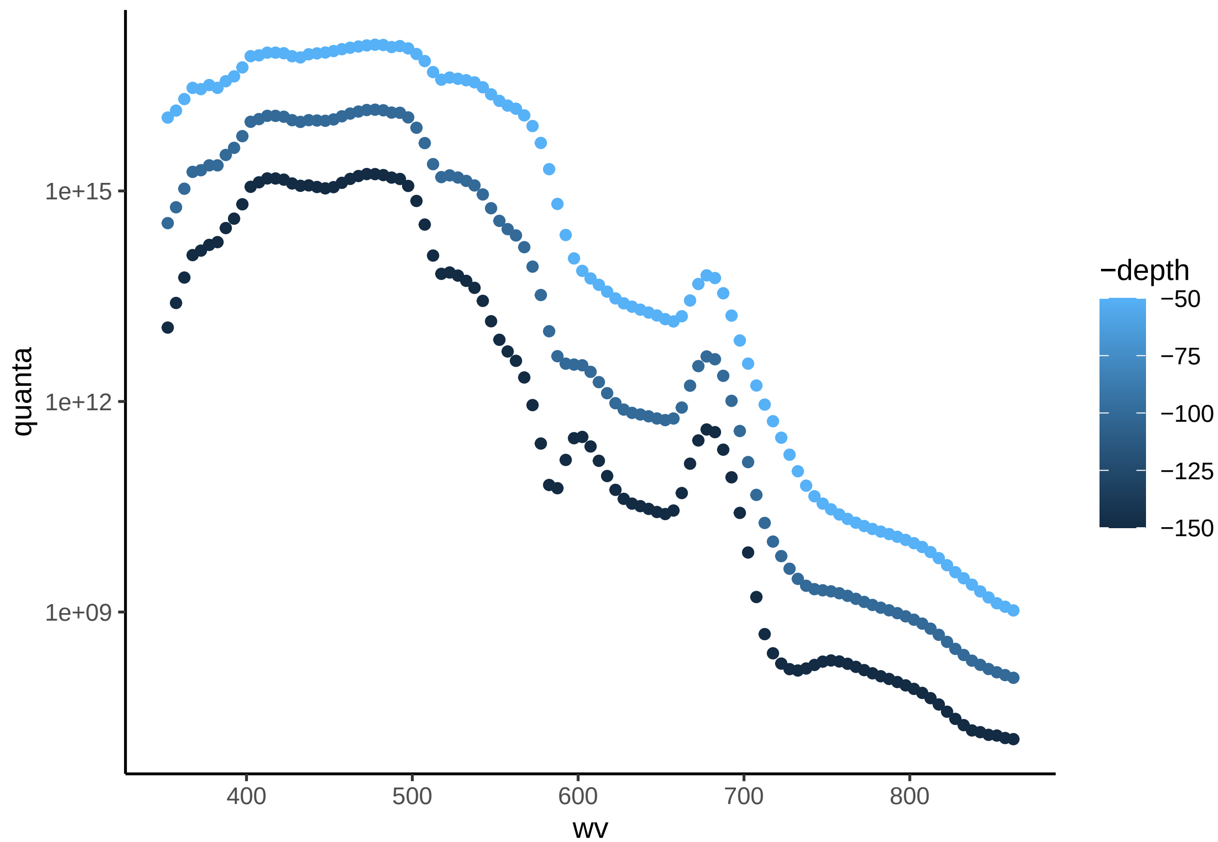
About 1% in the blue range - less elsewhere.

We gather these data at various depths into a single *depth* variable. The quanta is the irradiance (converted to moles) by Avogrado's number. We multiply it by 5 (the binwidth),as values are per nm in the specfied range.

```
E0_df %>% tidyr::gather(key="depth",value="irrad",-'wv') %>%
  group_by(depth,wv) %>%
  mutate(irrad_SD = round(sd(irrad),2),
         quanta    = (irrad / 1e6) * 6.02214076e23
         ) -> scalar_irrad_df
scalar_irrad_df$depth <- as.numeric(scalar_irrad_df$depth)
head(scalar_irrad_df)
```

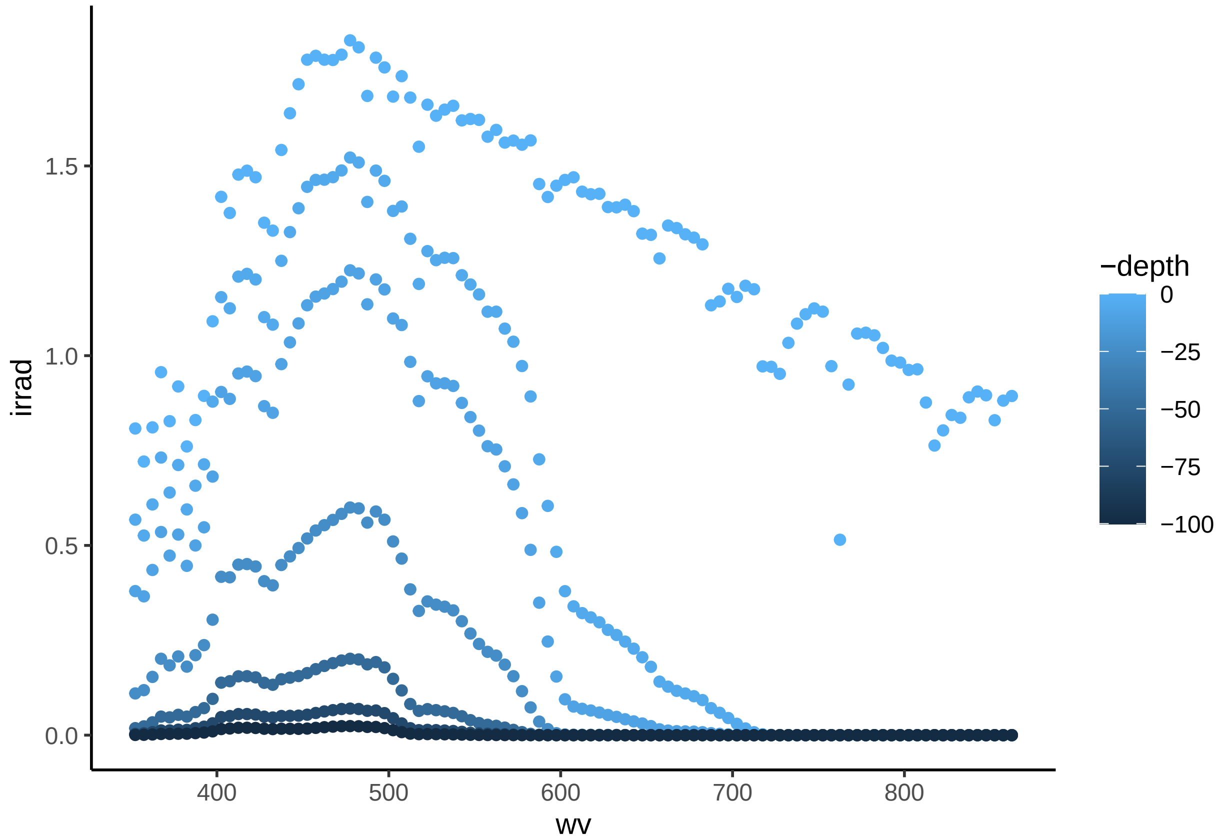
```
## # A tibble: 6 x 5
## # Groups:   depth, wv [6]
##        wv depth irrad irrad_SD  quanta
##     <dbl> <dbl> <dbl>    <dbl>   <dbl>
## 1  352.     0 0.808       NA 4.87e17
## 2  358.     0 0.721       NA 4.34e17
## 3  362.     0 0.811       NA 4.88e17
## 4  368.     0 0.956       NA 5.76e17
## 5  372.     0 0.827       NA 4.98e17
## 6  378.     0 0.919       NA 5.53e17
```

```
scalar_irrad_df %>% group_by(depth) %>%
  filter(depth %in% c(50,100,150)) %>%
 # filter(depth[depth%%2==0]) %>%
  ggplot(aes(x=wv,y=quanta,color=-depth)) + geom_point() + theme_classic() + scale_y_log10()
```



This is approximately in keeping with S. Johnsen's Optics of life Absorption chapter, which gives similar values for light at 500 nm (using irradiance with units masured in sq cm.)

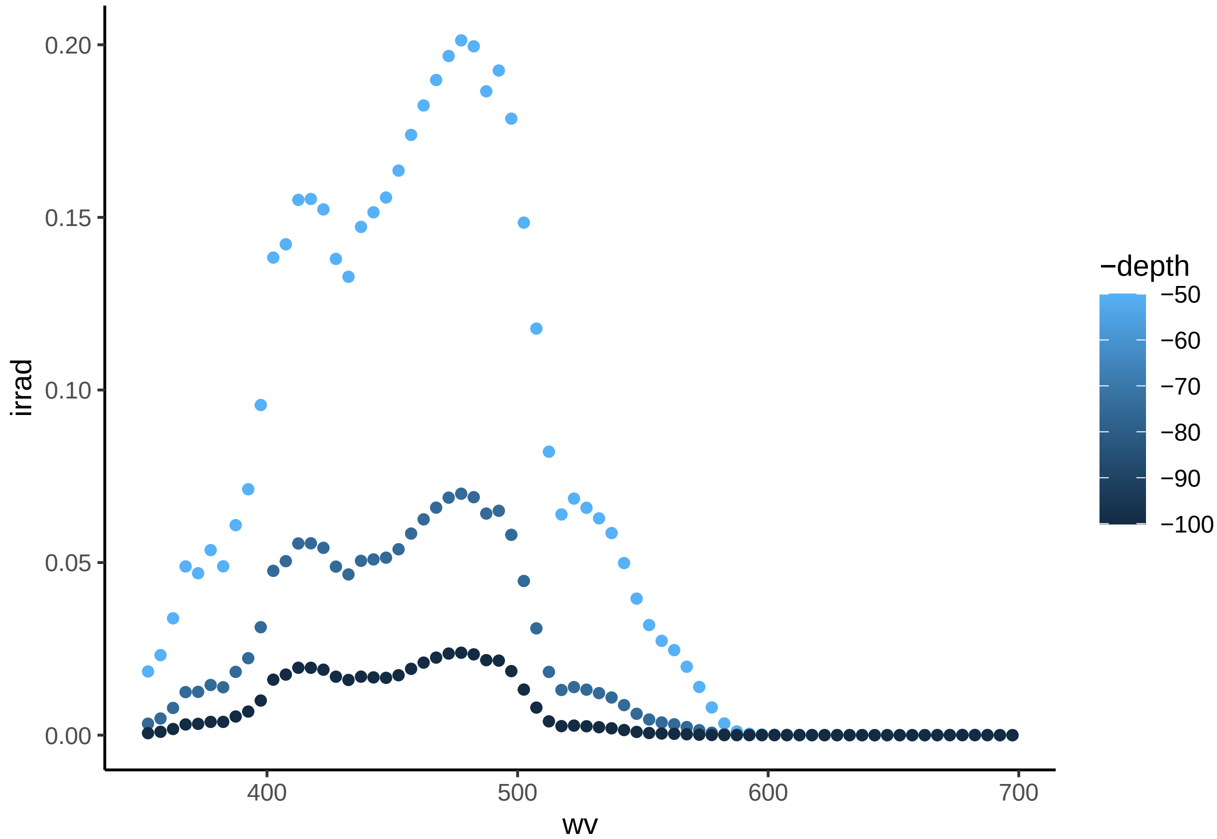
```
scalar_irrad_df %>% group_by(depth) %>%
  filter(depth %in% c(0,5,10,25,50,75,100)) %>%
 # filter(depth[depth%%2==0]) %>%
  ggplot(aes(x=wv,y=irrad,color=-depth)) + geom_point() + theme_classic() #+
```




```
  #scale_y_log10()
```

Focus on the deepest three conditions.

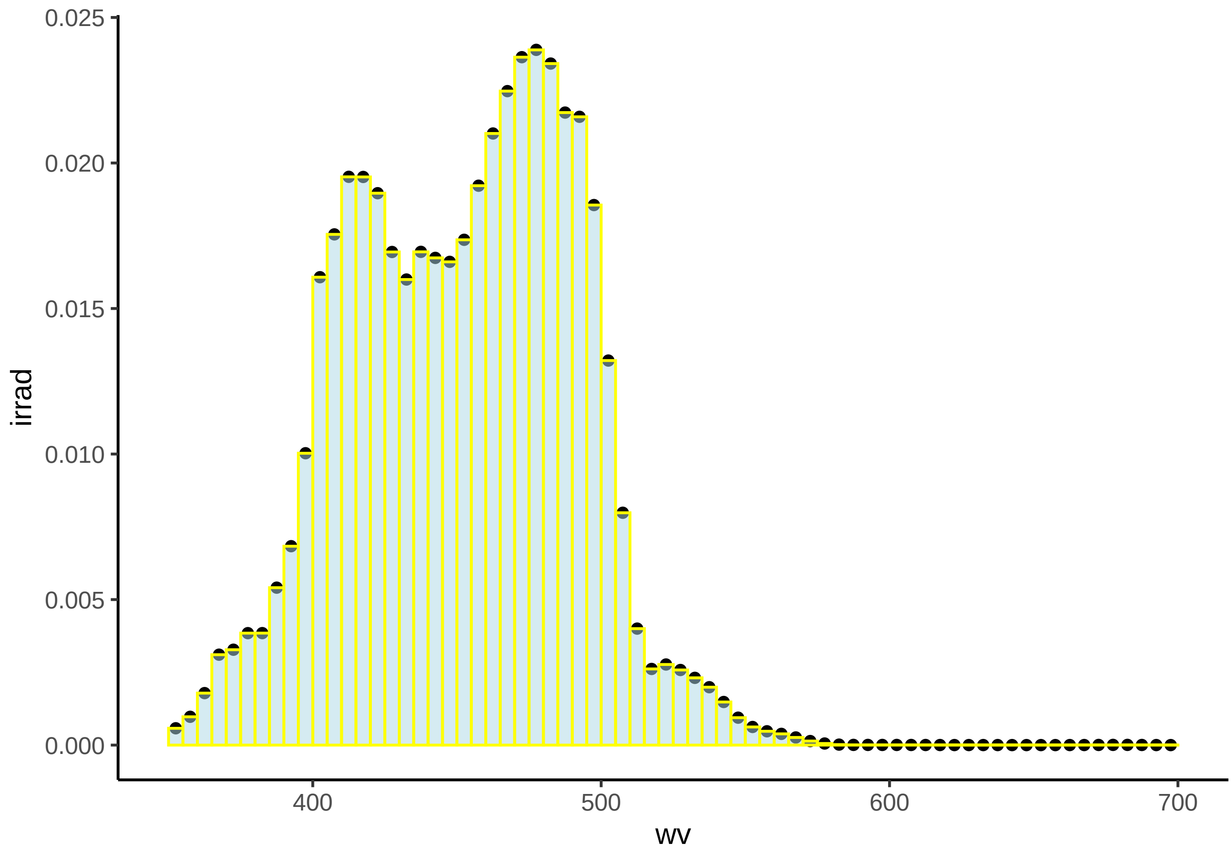
```
scalar_irrad_df %>% group_by(depth) %>%
  filter(between(wv,350,700)) %>%
  filter(depth %in% c(50,75,100)) %>%
 # filter(depth[depth%%2==0]) %>%
  ggplot(aes(x=wv,y=irrad,color=-depth)) + geom_point() + theme_classic() #+
```




```
  #scale_y_log10()
```

Get the relative decrement from 50 m to 75 m depth and apply this to 75 m depth to interpolate 100 m depth.

```
scalar_irrad_df %>% group_by(depth) %>%
  filter(wv < 700) %>%
  filter(depth %in% 100) %>%
 # filter(depth[depth%%2==0]) %>%
  ggplot(aes(x=wv,y=irrad)) + geom_point() +
  geom_col(width=5,alpha=.5,fill="lightblue",col="yellow") +
 # geom_smooth(method = "loess", span=0.1) +
  theme_classic()
```



There is actually still some power below 400 nm (into near UV).

```
print(paste0('Percentage of photons below 400nm at 100 m depth: ',
  signif(100 * sum(scalar_irrad_df$quanta[
    scalar_irrad_df$wv < 400 & scalar_irrad_df$depth==100]) /
    sum(scalar_irrad_df$quanta[scalar_irrad_df$depth==0 & scalar_irrad_df$wv < 400]),
  2),'%'))
```

```
## [1] "Percentage of photons below 400nm at 100 m depth: 0.46%"
```

A non-negligible amount of UV light remains at 100m. We will ignore these photons in finding a scalar value of irradiance that we can match with the PAR-sensitive (400-700 nm) irradiance meter at Stazione Zoologica Anton Dohrn.

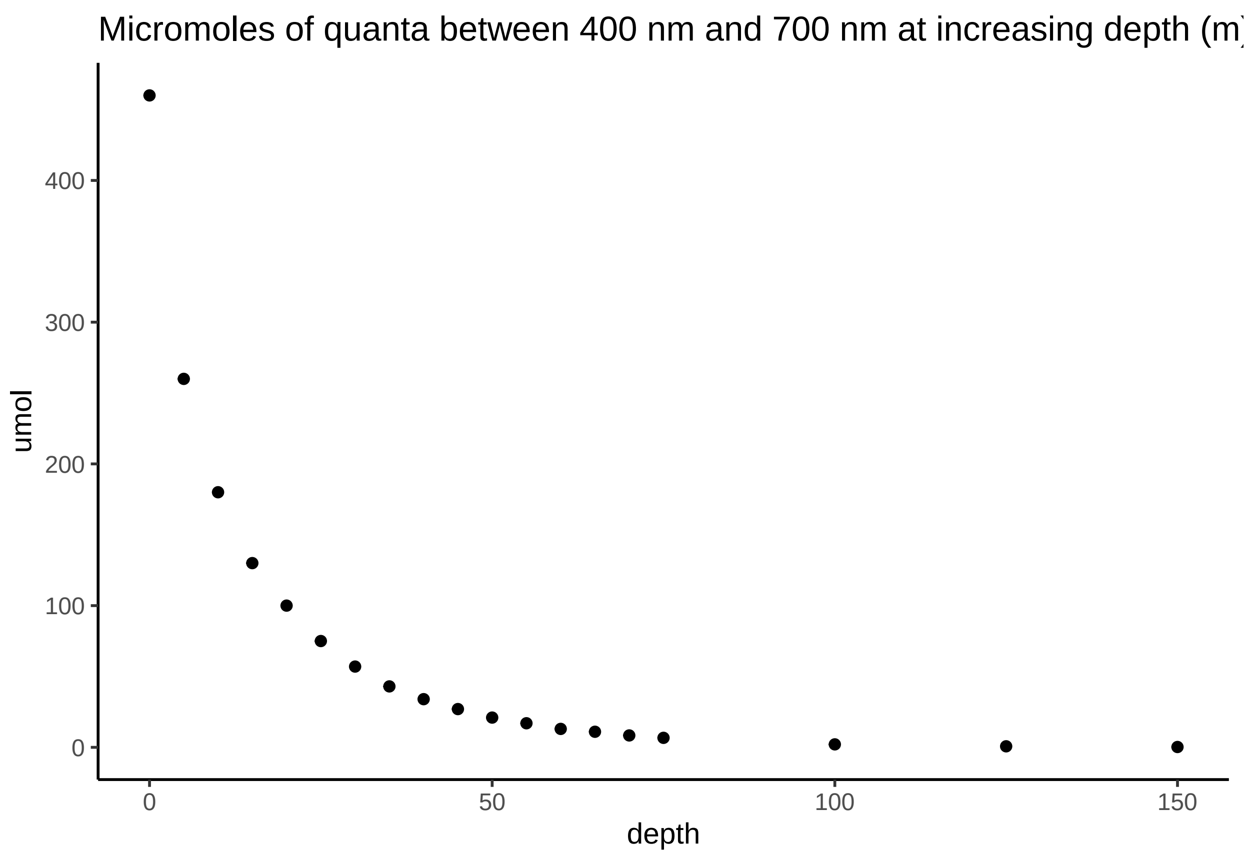
Sum values

How many moles of photons (400 - 700) are present in the light at various depths?

Multiply the values by 5 (as the binwidth is 5 nm and values are per nm in that range).

```
scalar_irrad_df %>% filter(between(wv,400,700)) %>%
  group_by(depth) %>%
  summarise(umol    = signif(5 * sum(irrad),2),
            umol_sd = signif(5 * sd(irrad),1),
            quanta  = signif(5 * sum(irrad/1e6)*6.02214076e23,3),
```

```
            ) -> irrad_sum_df
qplot(data=irrad_sum_df,depth,umol,main='Micromoles of quanta between 400 nm and 700 nm at increasing de
```



Light level at 5m depth The downwelling light at 5 m is:

```
print(paste(
  round(sum(5 * E0_df$`5`[E0_df$wv >= 400 & E0_df$wv <= 700]),1),
  'umol m^-2 s^-1.'
))
```

```
## [1] "262.9 umol m^-2 s^-1."
```

The downwelling light at 10 m is:

```
print(paste(
  round(sum(5 * E0_df$`10`[E0_df$wv >= 400 & E0_df$wv <= 700]),1),
  'umol m^-2 s^-1.'
))
```

```
## [1] "184.5 umol m^-2 s^-1."
```

At 75 m

```
print(paste(
  round(sum(5 * E0_df$`75`[E0_df$wv >= 400 & E0_df$wv <= 700]),1),
  'umol m^-2 s^-1.'
))
```

```
## [1] "6.7 umol m^-2 s^-1."
```

And at 100 m:

```
print(paste(
  round(sum(5 * E0_df$`100`[E0_df$wv >= 400 & E0_df$wv <= 700]),1),
  'umol m^-2 s^-1.'
))
```

```
## [1] "2.1 umol m^-2 s^-1."
```

Converting Spectral Irradiance to Lux

```
irrad2lux <- read.csv('spectral_irradiance_2_lux.csv')
head(irrad2lux)
```

```
##    wv y_bar_lambda
## 1 380      3.9e-05
## 2 390      1.2e-04
## 3 400      4.0e-04
## 4 410      1.2e-03
## 5 420      4.0e-03
## 6 430      1.2e-02
```

Planck's constant * speed of light / wavelength gives the energy of a photon in joules. This times the number of quanta per second gives the Wattage (1 W = 1 J / s).

```
h <- 6.62607015e-34 # Planck's constant
c <- 299792458e9    # speed of light in nm

scalar_irrad_df |> select(-irrad_SD) |>
  mutate(quanta_in_5nm_bin = quanta * 5) |>
  mutate(W_in_5nm_bin = quanta_in_5nm_bin * h * (c/wv) ) -> Watts_5nm_bin
```

Add new 10 nm bins.

```
bins_10nm <- seq(from=385,to=865,by=10)
Watts_5nm_bin %>%
  mutate(Wv10nm = bins_10nm[which.min(abs(bins_10nm - wv))]-5) -> Watts_5nm_bin
head(Watts_5nm_bin)
```

```
## # A tibble: 6 x 7
## # Groups:   depth, wv [6]
##      wv depth irrad  quanta quanta_in_5nm_bin W_in_5nm_bin Wv10nm
```

```
##    <dbl> <dbl> <dbl>   <dbl>              <dbl>        <dbl>  <dbl>
## 1  352.     0 0.808 4.87e17            2.43e18         1.37    380
## 2  358.     0 0.721 4.34e17            2.17e18         1.21    380
## 3  362.     0 0.811 4.88e17            2.44e18         1.34    380
## 4  368.     0 0.956 5.76e17            2.88e18         1.56    380
## 5  372.     0 0.827 4.98e17            2.49e18         1.33    380
## 6  378.     0 0.919 5.53e17            2.77e18         1.46    380
```

Sum the power into the 10 nm bins. The values were previously multiplied by 5 to reverse the averaging per nm and the totals per 5 nm bin were added to get totals per 10 nm bin. These are below divided by 10 to get per nm averages, i.e. spectral irradiance.

```
Watts_5nm_bin %>% group_by(Wv10nm,depth) %>%
  summarise(W_in_10nm_bin = sum(W_in_5nm_bin),
            W_in_10nm_per_nm = W_in_10nm_bin/10,
            quanta_in_10nm_bin = sum(quanta_in_5nm_bin)) -> Watts_10nm_bin
```

```
## `summarise()` has grouped output by 'Wv10nm'. You can override using the
## `.groups` argument.
```

```
head(Watts_10nm_bin)
```

```
## # A tibble: 6 x 5
## # Groups:   Wv10nm [1]
##   Wv10nm depth W_in_10nm_bin W_in_10nm_per_nm quanta_in_10nm_bin
##    <dbl> <dbl>         <dbl>            <dbl>              <dbl>
## 1    380     0         10.7             1.07             2.00e19
## 2    380     5          8.14            0.814            1.52e19
## 3    380    10          5.91            0.591            1.10e19
## 4    380    15          4.23            0.423            7.91e18
## 5    380    20          3.04            0.304            5.68e18
## 6    380    25          2.20            0.220            4.11e18
```

Taken from Sönke's book

> You can convert an irradiance spectrum E(λ) given in W/m2/nm to lux using:

$$lux = 637 \sum_{\lambda=380}^{\lambda=780} E(\lambda) * \overline{y}(\lambda) * \Delta\lambda$$

where $\overline{y}(\lambda)$ is the photopic (light-adapted) luminosity curve for humans (given in the table below). In other words, to calculate lux, you multiply your irradiance spectrum (first binned into 10 nm intervals) by the luminosity curve and by $\Delta\lambda$ (which is **10 nm** in this case) and then add up all the products. Then multiply this final sum by 673. This is most easily done in a spreadsheet program. Remember that your irradiance must be in watts, since that's how the luminosity curve was created. The values of (λ) below 400 nm and above 700 nm are small, so don't worry if you don't have irradiance data for this range. The table is in *'spectral_irradiance_2_lux.csv'*.

The human photometric conversions are in 10 nm bins, whereas the Neapolitan values are in 5 nm bins. Therefore, average the Naples values into 10 nm bins, get as Watts (all above) and calculate.

Get only relevant depths and wvs

```r
Watts_10nm_bin %>% filter(depth == 0, between(Wv10nm,380,780)) -> Watts_10nm_0m
Watts_10nm_0m$y_bar_lambda <- irrad2lux$y_bar_lambda
head(Watts_10nm_0m)
```

```
## # A tibble: 6 x 6
## # Groups:   Wv10nm [6]
##   Wv10nm depth W_in_10nm_bin W_in_10nm_per_nm quanta_in_10nm_bin y_bar_lambda
##    <dbl> <dbl>         <dbl>            <dbl>              <dbl>        <dbl>
## 1    380     0         10.7             1.07             2.00e19     0.000039
## 2    390     0          3.00            0.300            5.98e18     0.00012
## 3    400     0          4.13            0.413            8.41e18     0.0004
## 4    410     0          4.27            0.427            8.93e18     0.0012
## 5    420     0          3.97            0.397            8.49e18     0.004
## 6    430     0          3.95            0.395            8.65e18     0.012
```

```r
Watts_10nm_0m |>
  mutate(lux = 637*(W_in_10nm_per_nm)*(y_bar_lambda*10),
         klx = lux / 1000) -> Watts_10nm_0m
head(Watts_10nm_0m)
```

```
## # A tibble: 6 x 8
## # Groups:   Wv10nm [6]
##   Wv10nm depth W_in_10nm_bin W_in_10nm_per_nm quanta_in_10nm_bin y_bar_lambda
##    <dbl> <dbl>         <dbl>            <dbl>              <dbl>        <dbl>
## 1    380     0         10.7             1.07             2.00e19     0.000039
## 2    390     0          3.00            0.300            5.98e18     0.00012
## 3    400     0          4.13            0.413            8.41e18     0.0004
## 4    410     0          4.27            0.427            8.93e18     0.0012
## 5    420     0          3.97            0.397            8.49e18     0.004
## 6    430     0          3.95            0.395            8.65e18     0.012
## # ... with 2 more variables: lux <dbl>, klx <dbl>
```

```r
sum(Watts_10nm_0m$klx)
```

```
## [1] 22.85042
```

This value is slightly below what would be expected from direct sunlight.