

Limnología Aplicada: Metabolismo de ecosistemas acuáticos.

Parte I

Jorge Juan Montes Pérez

12 de mayo de 2020

Introducción

En el medio acuático tienen lugar todos los procesos metabólicos a través de los cuales la vida ha encontrado la manera de perpetuar su existencia modificando las sustancias que le rodean. La diversidad de las rutas a través de las cuales los organismos son capaces de obtener energía para realizar todos los procesos vitales es inabarcable en una práctica y, además, no es el objetivo final de esta. Sin embargo, las dos principales rutas que dominan el mundo aeróbico, debido a su gran eficiencia frente a las demás, son la fotosíntesis oxigénica y la respiración aeróbica,¹ el resto de rutas encuentra su máxima relevancia en aquellos nichos en los que se dan las condiciones adecuadas y no son posible otros mecanismos más eficientes de obtención de energía. Por lo tanto, en un sistema acuático bien oxigenado los dos principales procesos que tienen un mayor impacto en el ecosistema son la fotosíntesis y la respiración. Del balance entre ambos procesos puede depender la calidad del agua, tornándose anóxica, si prima la respiración por encima de los procesos de producción primaria, y comprometiendo muchas formas de vida. Además, ambos procesos juegan un papel relevante en el ciclo del carbono, retirando CO₂ del entorno mediante la fotosíntesis (PP) o liberando CO₂ a través de la respiración.

Al igual que a nivel de organismo o de individuo, el metabolismo puede ser interpretado a nivel de ecosistema como el conjunto de procesos metabólicos que tienen lugar en dicho ecosistema. Como hemos dicho anteriormente, en un sistema acuático bien oxigenado, fotosíntesis y respiración son los procesos más importantes que determinan el metabolismo del ecosistema y su papel en los ciclos biogeoquímicos globales.

Si queremos estimar el metabolismo aeróbico de un lago, los cambios en la concentración de oxígeno disuelto (OD) es una de las variables más utilizadas. Esta puede ser medida en botellas de incubación y monitorizar el cambio en la concentración de O₂ durante un periodo de tiempo determinado y en un volumen de agua concreto. Esto ofrece la ventaja de controlar las condiciones ambientales (temperatura, irradiancia, etc) y poder realizar replicas. Sin embargo, el “efecto contenedor” conlleva ciertas limitaciones a la hora de escalar a nivel de ecosistema. Otra manera puede ser medir los cambios en la concentración de oxígeno directamente en la masa de agua (“aguas libres”), de este modo el escalado a nivel del conjunto de la masa es más realista, no sin ciertas limitaciones e incertidumbres (Staehr and Sand-Jensen 2007, Mccutchan et al. (1998), Hanson et al. (2008), Staehr et al. (2010)). Sabemos que la respiración y la producción primaria afectan directamente a la concentración de OD, consumiendo oxígeno en el primer caso y produciendolo en el segundo. Sin embargo, hay otros factores físicos que pueden afectar a la concentración de OD (Ver figura). Por lo tanto, podríamos definir el cambio en la concentración de oxígeno con la siguiente ecuación:

$$\frac{\Delta OD}{\Delta t} = GPP - R - At_{ex} - E$$

Donde, GPP es la producción primaria bruta, R es la respiración, At_{ex} es el intercambio gaseoso con la atmósfera y E serían otros factores, de menor magnitud, que afectan a la concentración de OD como, por ejemplo, la oxidación fotoquímica de la materia orgánica o advección.

¹A partir de ahora, siempre que nos refiramos a la fotosíntesis o a la respiración a secas nos estamos refiriendo a la fotosíntesis oxigénica y la respiración aeróbica.

Por lo tanto, si medimos los cambios que se producen en la concentración de OD durante un día completo en un lago, y podemos calcular el intercambio gaseoso de O_2 que se produce entre la masa de agua y la atmósfera, podremos estimar su metabolismo. Por la noche, el cambio en la concentración de OD, no producto del intercambio gaseoso, será debido a la respiración. Durante el día, sin embargo, este cambio será debido a la producción neta (NEP). Conociendo que $NEP = GPP - R$ y habiendo calculado la respiración, podemos averiguar cual es la producción primaria bruta del sistema.

Este modelo tiene ciertas asunciones que conviene tener presente:

1. Los cambios en la concentración de OD son el resultado del balance entre producción fotosintética y respiración así como del intercambio con la atmósfera. Es decir, asume que E es despreciable.
2. La producción primaria tiene lugar solo durante el día.
3. La tasa de respiración durante la noche es la misma que durante el día.

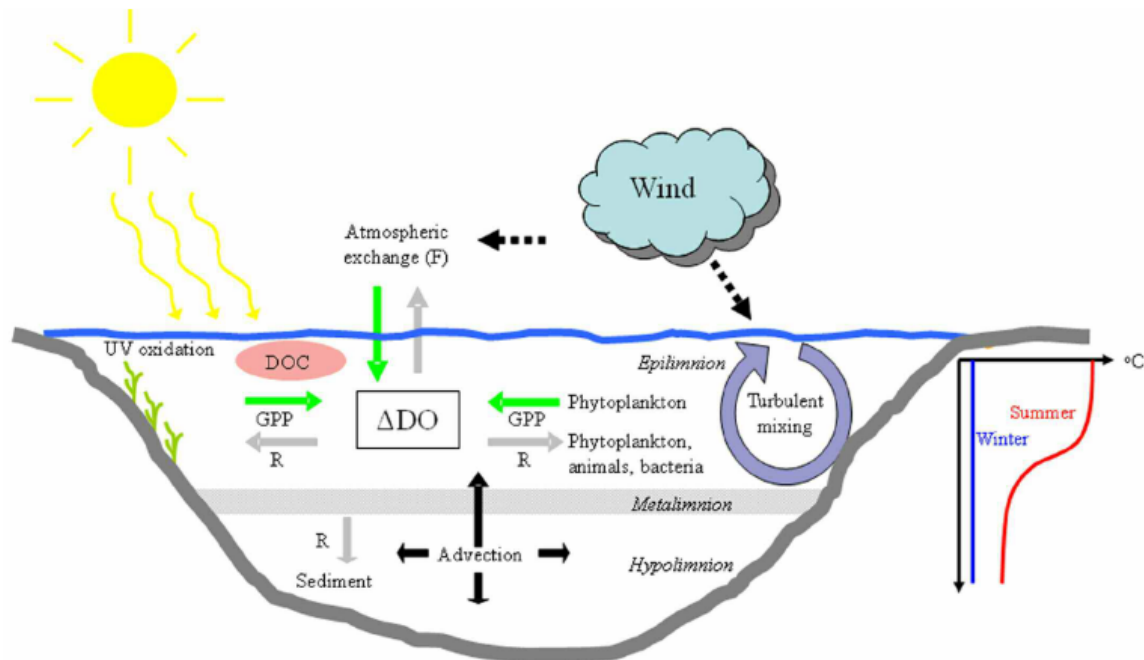


Figure 1: Modelo conceptual de los principales factores físicos y biológicos que afecta a las variaciones en la concentración de oxígeno disuelto en un lago. Imagen extraída de Staehr et al. (2010).

Estructura de la práctica

1. Obtener los datos con los que vamos a trabajar. Usaremos la red The Global Lake Ecological Observatory Network (GLEON). Esta red pone a nuestra disposición una amplia cantidad de datos de monitoreo de alta frecuencia (HFM) de distintos lugares del mundo.
2. Familiarizarnos con los paquetes y funciones básicos de R que nos facilitan explorar y trabajar con grandes tablas de datos.
3. Estimar el metabolismo de un lago usando el paquete de R LakeMetabolizer.
4. Explorar y representar los resultados obtenidos.

Descargar los datos de la red GLEON

Para ello, visitamos la página de la red GLEON y nos vamos al apartado de datos. En esta sección podemos encontrar la política de datos de GLEON, básicamente se apuesta por una ciencia colaborativa en la que

Pues bien, para este práctica vamos a trabajar, en concreto, con datos del lago Trout. Así que utilizando el buscador que más sea de vuestro agrado lanzamos la siguiente búsqueda: *ecosystem metabolism lake trout*. Entre los resultados obtenidos (hay bastante información como podéis observar), vamos a seleccionar los datos derivados del proyecto North Temperate Lakes Long Term Ecological Research (NTL-LTER) que nos ofrecen datos meteorológicos y de oxígeno disuelto desde 2004 hasta la actualidad. Si no pudierais encontrarlos, podéis pinchar aquí: North Temperate Lakes LTER: High Frequency Meteorological and Dissolved Oxygen Data - Trout Lake Buoy 2004 - current.

Disponemos de dos resoluciones temporales de datos, una diaria y otra horaria. Si queremos estimar el metabolismo no nos sirven los datos diarios. Para descargar los datos tenemos dos opciones:

- Esta es la pinta que tiene el script:

3

```

dt1 <-read.csv(infile1,header=F
              ,skip=1
              ,sep=","
              ,quot="'"
              , col.names=c(
                  "sampledate",
                  "year4",
                  "month",
                  "daynum",
                  "avg_air_temp",
                  "flag_avg_air_temp",
                  "avg_rel_hum",
                  "flag_avg_rel_hum",
                  "avg_wind_speed",
                  "flag_avg_wind_speed",
                  "avg_wind_dir",
                  "flag_avg_wind_dir",
                  "avg_do_raw",
                  "flag_avg_do_raw",
                  "avg_do_sat",
                  "flag_avg_do_sat",
                  "avg_do_wtemp",
                  "flag_avg_do_wtemp",
                  "avg_barom_pres_mbar",
                  "flag_avg_barom_pres_mbar",
                  "avg_par",
                  "flag_avg_par"      ), check.names=TRUE)

```

Esta primera parte del script es para descargar los datos diarios, los cuales dijimos que no nos interesaban, por lo tanto vamos más abajo en el script (entre las líneas 110-207) para descargar los datos por hora. Este es aspecto que tiene (salvo que me he tomado el tiempo de comentar algunas líneas):

```

inUrl2 <- "https://pasta.ltnet.net/package/data/eml/knb-lter-ntl/117/38/7f36b24e62c5798b16517e9dd85f
infile2 <- tempfile() #Crea un objeto temporal donde posteriormente guardar los datos
download.file(inUrl2,infile2,method="curl") #Decarga los datos al objeto temporal (A vosotros os debe a

```

```

dt2 <-read.csv(infile2,header=F #Crea un objeto donde "vuelca los datos descargados"
              ,skip=1
              ,sep=","
              ,quot="'"
              , col.names=c(
                  "sampledate",      #Asigna nombre a las columnas
                  "year4",
                  "month",
                  "daynum",
                  "hour",
                  "avg_air_temp",
                  "flag_avg_air_temp",
                  "avg_rel_hum",
                  "flag_avg_rel_hum",
                  "avg_wind_speed",
                  "flag_avg_wind_speed",
                  "avg_wind_dir",

```

```

"flag_avg_wind_dir",
"avg_do_raw",
"flag_avg_do_raw",
"avg_do_sat",
"flag_avg_do_sat",
"avg_do_wtemp",
"flag_avg_do_wtemp",
"avg_barom_pres_mbar",
"flag_avg_barom_pres_mbar",
"avg_par",
"flag_avg_par"    ), check.names=TRUE)

```

Este trocito de código que sigue es para corregir algún problema de formato que se haya podido introducir debido a algún error en la base de datos.

```

# Fix any interval or ratio columns mistakenly read in as nominal and nominal columns read as numeric or

# attempting to convert dt2$sampledate dateTime string to R date structure (date or POSIXct)
tmpDateFormat<-"%Y-%m-%d"
tmp2sampledate<-as.Date(dt2$sampledate,format=tmpDateFormat)
# Keep the new dates only if they all converted correctly
if(length(tmp2sampledate) == length(tmp2sampledate[!is.na(tmp2sampledate)])){dt2$sampledate <- tmp2samp
rm(tmpDateFormat,tmp2sampledate)
if (class(dt2$year4)=="factor") dt2$year4 <-as.numeric(levels(dt2$year4))[as.integer(dt2$year4) ]
if (class(dt2$month)=="factor") dt2$month <-as.numeric(levels(dt2$month))[as.integer(dt2$month) ]
if (class(dt2$daynum)=="factor") dt2$daynum <-as.numeric(levels(dt2$daynum))[as.integer(dt2$daynum) ]
if (class(dt2$avg_air_temp)=="factor") dt2$avg_air_temp <-as.numeric(levels(dt2$avg_air_temp))[as.integer(
if (class(dt2$flag_avg_air_temp)!="factor") dt2$flag_avg_air_temp<- as.factor(dt2$flag_avg_air_temp)
if (class(dt2$avg_rel_hum)=="factor") dt2$avg_rel_hum <-as.numeric(levels(dt2$avg_rel_hum))[as.integer(
if (class(dt2$flag_avg_rel_hum)!="factor") dt2$flag_avg_rel_hum<- as.factor(dt2$flag_avg_rel_hum)
if (class(dt2$avg_wind_speed)=="factor") dt2$avg_wind_speed <-as.numeric(levels(dt2$avg_wind_speed))[as
if (class(dt2$flag_avg_wind_speed)!="factor") dt2$flag_avg_wind_speed<- as.factor(dt2$flag_avg_wind_spe
if (class(dt2$avg_wind_dir)=="factor") dt2$avg_wind_dir <-as.numeric(levels(dt2$avg_wind_dir))[as.integ
if (class(dt2$flag_avg_wind_dir)!="factor") dt2$flag_avg_wind_dir<- as.factor(dt2$flag_avg_wind_dir)
if (class(dt2$avg_do_raw)=="factor") dt2$avg_do_raw <-as.numeric(levels(dt2$avg_do_raw))[as.integer(dt2
if (class(dt2$flag_avg_do_raw)!="factor") dt2$flag_avg_do_raw<- as.factor(dt2$flag_avg_do_raw)
if (class(dt2$avg_do_sat)=="factor") dt2$avg_do_sat <-as.numeric(levels(dt2$avg_do_sat))[as.integer(dt2
if (class(dt2$flag_avg_do_sat)!="factor") dt2$flag_avg_do_sat<- as.factor(dt2$flag_avg_do_sat)
if (class(dt2$avg_do_wtemp)=="factor") dt2$avg_do_wtemp <-as.numeric(levels(dt2$avg_do_wtemp))[as.integ
if (class(dt2$flag_avg_do_wtemp)!="factor") dt2$flag_avg_do_wtemp<- as.factor(dt2$flag_avg_do_wtemp)
if (class(dt2$avg_barom_pres_mbar)=="factor") dt2$avg_barom_pres_mbar <-as.numeric(levels(dt2$avg_barom
if (class(dt2$flag_avg_barom_pres_mbar)!="factor") dt2$flag_avg_barom_pres_mbar<- as.factor(dt2$flag_av
if (class(dt2$avg_par)=="factor") dt2$avg_par <-as.numeric(levels(dt2$avg_par))[as.integer(dt2$avg_par)
if (class(dt2$flag_avg_par)!="factor") dt2$flag_avg_par<- as.factor(dt2$flag_avg_par)

# Convert Missing Values to NA for non-dates

```

Este último trozo es simplemente para ver la estructura de los datos y un resumen de cada una de la variables. Este trozo no es necesario que lo ejecutéis.

```

# Here is the structure of the input data frame:
str(dt2)

```

```

## 'data.frame':    55255 obs. of  23 variables:
## $ sampledate      : Date, format: "2004-05-21" "2004-05-21" ...

```

```
## $ year4 : int 2004 2004 2004 2004 2004 2004 2004 2004 2004 2004 ...
## $ month : int 5 5 5 5 5 5 5 5 5 5 ...
## $ daynum : int 142 142 142 142 142 142 142 142 142 143 ...
## $ hour : int 15 16 17 18 19 20 21 22 23 0 ...
## $ avg_air_temp : num 0 0 0 0 0 0 0 0 0 0 ...
## $ flag_avg_air_temp : Factor w/ 4 levels "", "C", "D", "H": 2 2 2 2 2 2 2 2 2 2 ...
## $ avg_rel_hum : num NA NA NA NA NA NA NA NA NA NA ...
## $ flag_avg_rel_hum : Factor w/ 3 levels "", "C", "D": 2 2 2 2 2 2 2 2 2 2 ...
## $ avg_wind_speed : num NA NA NA NA NA NA NA NA NA NA ...
## $ flag_avg_wind_speed : Factor w/ 4 levels "", "C", "D", "H": 2 2 2 2 2 2 2 2 2 2 ...
## $ avg_wind_dir : num 226 246 256 295 290 ...
## $ flag_avg_wind_dir : Factor w/ 4 levels "", "C", "D", "H": 1 1 1 1 1 1 1 1 1 1 ...
## $ avg_do_raw : num 11.1 11.1 11.1 11.1 11.1 ...
## $ flag_avg_do_raw : Factor w/ 5 levels "", "C", "D", "H", "...: 1 1 1 1 1 1 1 1 1 1 ...
## $ avg_do_sat : num 99.5 99.5 99.5 99.5 99.4 ...
## $ flag_avg_do_sat : Factor w/ 5 levels "", "C", "D", "H", "...: 1 1 1 1 1 1 1 1 1 1 ...
## $ avg_do_wtemp : num 10.5 10.5 10.5 10.5 10.4 ...
## $ flag_avg_do_wtemp : Factor w/ 5 levels "", "C", "D", "H", "...: 1 1 1 1 1 1 1 1 1 1 ...
## $ avg_barom_pres_mbar : num NA NA NA NA NA NA NA NA NA NA ...
## $ flag_avg_barom_pres_mbar: Factor w/ 0 levels: NA NA NA NA NA NA NA NA NA NA ...
## $ avg_par : num 207.8 76.2 69 67.7 5.7 ...
## $ flag_avg_par : Factor w/ 5 levels "", "C", "D", "F", "...: 4 4 4 4 4 4 4 4 4 4 ...

attach(dt2)
# The analyses below are basic descriptions of the variables. After testing, they should be replaced.

summary(sampledate)

##      Min.      1st Qu.      Median      Mean      3rd Qu.      Max.
## "2004-05-21" "2006-10-02" "2010-08-22" "2010-12-09" "2014-07-15" "2018-11-12"

summary(year4)

##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.
##      2004    2006    2010    2010    2014    2018

summary(month)

##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.
##      1.00   7.00   8.00   7.98   9.00  11.00

summary(daynum)

##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.
##      31.0  190.0  233.0  227.5  272.0  334.0

summary(hour)

##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.
##       0.0    5.5   11.0   11.5   18.0   23.0

summary(avg_air_temp)

##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.
##     -12.24    6.72   15.11   13.26   19.77   31.68

summary(flag_avg_air_temp)

##           C           D           H
```

```
## 48513 6727 14 1
```

```
summary(avg_rel_hum)
```

```
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
##  26.79   64.60   77.36   74.74   86.68  103.60   6928
```

```
summary(flag_avg_rel_hum)
```

```
##           C      D
## 48487 6752 16
```

```
summary(avg_wind_speed)
```

```
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
##    0.000   2.190   3.600   3.904   5.100 1000.000  10785
```

```
summary(flag_avg_wind_speed)
```

```
##           C      D      H
## 47779 7238 233 5
```

```
summary(avg_wind_dir)
```

```
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
##    0.0   148.6   211.7   210.0   284.9  1000.0   7324
```

```
summary(flag_avg_wind_dir)
```

```
##           C      D      H
## 47991 6861 398 5
```

```
summary(avg_do_raw)
```

```
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## -7999.000   8.690   9.250   4.306  10.330   42.120   7550
```

```
summary(flag_avg_do_raw)
```

```
##           C      D      H      0
## 47888 3771 1700 1887 9
```

```
summary(avg_do_sat)
```

```
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## -7999.00   92.89   99.93   94.99  104.65  133.97   7552
```

```
summary(flag_avg_do_sat)
```

```
##           C      D      H      0
## 48678 3758 1718 1092 9
```

```
summary(avg_do_wtemp)
```

```
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## -7999.00   14.05   19.12   12.21   21.64   59.99   7705
```

```
summary(flag_avg_do_wtemp)
```

```
##           C      D      H      0
## 51559 2503 1087 97 9
```

```
summary(avg_barom_pres_mbar)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.     NA's  
##    931.0   956.2   960.0   959.9   964.0   977.0   44268
```

```
summary(flag_avg_barom_pres_mbar)
```

```
## NA's  
## 55255
```

```
summary(avg_par)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.     NA's  
##    -0.67    0.00   47.39  295.80  471.45 2024.52   19557
```

```
summary(flag_avg_par)
```

```
##           C      D      F      H  
## 43248  5235    29  6722    21
```

```
detach(dt2)
```

Bien, una vez ejecutado el script, ya debemos tener los datos en nuestro entorno de RStudio en un tipo de objeto denominado data.frame (básicamente una tabla de datos). Vamos a ver que pinta tienen:

```
head(dt2)
```

```
##      sampledate year4 month daynum hour avg_air_temp flag_avg_air_temp avg_rel_hum  
## 1 2004-05-21 2004     5   142    15           0              C             NA  
## 2 2004-05-21 2004     5   142    16           0              C             NA  
## 3 2004-05-21 2004     5   142    17           0              C             NA  
## 4 2004-05-21 2004     5   142    18           0              C             NA  
## 5 2004-05-21 2004     5   142    19           0              C             NA  
## 6 2004-05-21 2004     5   142    20           0              C             NA  
##      flag_avg_rel_hum avg_wind_speed flag_avg_wind_speed avg_wind_dir  
## 1                   C              NA              C      226.25  
## 2                   C              NA              C      246.27  
## 3                   C              NA              C      255.92  
## 4                   C              NA              C      294.78  
## 5                   C              NA              C      289.70  
## 6                   C              NA              C      288.42  
##      flag_avg_wind_dir avg_do_raw flag_avg_do_raw avg_do_sat flag_avg_do_sat  
## 1                   11.09           99.50  
## 2                   11.10           99.50  
## 3                   11.11           99.50  
## 4                   11.11           99.50  
## 5                   11.11           99.42  
## 6                   11.09           99.18  
##      avg_do_wtemp flag_avg_do_wtemp avg_barom_pres_mbar flag_avg_barom_pres_mbar  
## 1          10.53              NA              <NA>  
## 2          10.52              NA              <NA>  
## 3          10.49              NA              <NA>  
## 4          10.48              NA              <NA>  
## 5          10.45              NA              <NA>  
## 6          10.43              NA              <NA>  
##      avg_par flag_avg_par  
## 1    207.85      F  
## 2     76.17      F
```



```
## 3    68.95      F
## 4    67.72      F
## 5     5.70      F
## 6     0.00      F
```

Por último, los vamos a guardar en la carpeta “Datos_descargados” que hemos creado..

```
write.csv(dt2, "./Datos_descargados/Datos_Trout.csv", row.names = FALSE)
```

Bibliografía

Hanson, P. C., S. R. Carpenter, N. Kimura, C. Wu, S. P. Cornelius, and T. K. Kratz. 2008. Evaluation of metabolism models for free-water dissolved oxygen methods in lakes. *Limnology and Oceanography: Methods* **6**: 454–465.

Mccutchan, J. H., W. M. Lewis, and J. F. Saunders. 1998. Uncertainty in the estimation of stream metabolism from open-channel oxygen concentrations. *Journal of the North American Benthological Society* **17**: 155–164.

Staehr, P. a, D. Bade, G. R. Koch, C. Williamson, P. Hanson, J. J. Cole, and T. Kratz. 2010. Lake metabolism and the diel oxygen technique: State of the science. *Limnology and Oceanography: Methods* **8**: 628–644.

Staehr, P. A., and K. Sand-Jensen. 2007. Temporal dynamics and regulation of lake metabolism. *Limnology and Oceanography* **52**: 108–120.