R-package **marelac** : utilities for the MArine, Riverine, Estuarine, LAcustrine and Coastal sciences

Karline Soetaert
NIOO-CEME
The Netherlands

Thomas Petzoldt
Technische Universität Dresden
Germany

Filip Meysman
VUB
Belgium

Abstract

Rpackage **marelac** (Soetaert and Meysman 2008) contains chemical and physical constants and functions, routines for unit conversion, and other utilities useful for MArine, Riverine, Estuarine, LAcustrine and Coastal sciences.

Keywords: marine, riverine, estuarine, lacustrine, coastal science, utilities, constants, R.

1. Introduction

R-package **marelac** has been designed as a tool for use by scientists working in the MArine, Riverine, Estuarine, LAcustrine and Coastal sciences.

It contains:

- chemical and physical constants, e.g. atomic weights, gas constants.
- conversion factors, e.g. gram to mol to liter conversions.
- functions, e.g. to estimate concentrations of conservative substances as a function of salinity, ...

2. constants

2.1. AtomicWeight

```
> AtomicWeight

$H
[1] 1.00794

$He
[1] 4.002602
```

```
$Li
[1] 6.941

$Be
[1] 9.012182

$B
[1] 10.811

$C
[1] 12.0107

$N
[1] 14.0067

$O
[1] 15.9994

$F
[1] 18.99840

$Ne
[1] 20.1797

$Na
[1] 22.98977

$Mg
[1] 24.305

$Al
[1] 26.98154

$Si
[1] 28.0855

$P
[1] 30.97376

$S
[1] 32.065

$Cl
[1] 35.453

$Ar
```

```
[1] 39.948

$K
[1] 39.0983

$Ca
[1] 40.078

$Sc
[1] 44.95591

$Ti
[1] 47.867

$V
[1] 50.9415

$Cr
[1] 51.9961

$Mn
[1] 54.93805

$Fe
[1] 55.845

$Co
[1] 58.9332

$Ni
[1] 58.6934

$Cu
[1] 63.546

$Zn
[1] 65.409

$Ga
[1] 69.723

$Ge
[1] 72.64

$As
[1] 74.9216
```

```
$Se
[1] 78.96

$Br
[1] 79.904

$Kr
[1] 83.798

$Rb
[1] 85.4678

$Sr
[1] 87.62

$Y
[1] 88.90585

$Zr
[1] 91.224

$Nb
[1] 92.90638

$Mo
[1] 95.94

$Tc
[1] NA

$Ru
[1] 101.07

$Rh
[1] 102.9055

$Pd
[1] 106.42

$Ag
[1] 107.8682

$Cd
[1] 112.411

$In
[1] 114.818
```

```
$Sn
[1] 118.71

$Sb
[1] 121.76

$Te
[1] 127.6

$I
[1] 126.9045

$Xe
[1] 131.293

$Cs
[1] 132.9055

$Ba
[1] 137.327

$La
[1] 138.9055

$Ce
[1] 140.116

$Pr
[1] 140.9076

$Nd
[1] 144.242

$Pm
[1] NA

$Sm
[1] 150.36

$Eu
[1] 151.964

$Gd
[1] 157.25

$Tb
```

```
[1] 158.9254

$Dy
[1] 162.5

$Ho
[1] 164.9303

$Er
[1] 167.259

$Tm
[1] 168.9342

$Yb
[1] 173.04

$Lu
[1] 174.967

$Hf
[1] 178.49

$Ta
[1] 180.9479

$W
[1] 183.84

$Re
[1] 186.207

$Os
[1] 190.23

$Ir
[1] 192.217

$Pt
[1] 195.084

$Au
[1] 196.9666

$Hg
[1] 200.59
```

```
$Tl
[1] 204.3833

$Pb
[1] 207.2

$Bi
[1] 208.9804

$Po
[1] NA

$At
[1] NA

$Rn
[1] NA

$Fr
[1] NA

$Ra
[1] NA

$Ac
[1] NA

$Th
[1] 232.0381

$Pa
[1] 231.0359

$U
[1] 238.0289

$Np
[1] NA

$Pu
[1] NA

$Am
[1] NA

$Cm
[1] NA
```

```
$Bk
[1] NA

$Cf
[1] NA

$Es
[1] NA

$Fm
[1] NA

$Md
[1] NA

$No
[1] NA

$Lr
[1] NA

$Rf
[1] NA

$Db
[1] NA

$Sg
[1] NA

$Bh
[1] NA

$Hs
[1] NA

$Mt
[1] NA

$Ds
[1] NA

$Rg
[1] NA

> AtomicWeight$H
```

```
[1] 1.00794
```

```
> (W_H2O<- with (AtomicWeight, 2*H + O))
```

```
[1] 18.01528
```

2.2. Constants

```
> data.frame(cbind(acronym=names(Constants),
+              matrix(ncol=3,byrow=TRUE,data=unlist(Constants),
+              dimnames=list(NULL,c("value","units","description")))))
```

```
  acronym          value       units               description
1       g            9.8        m/s2      gravity acceleration
2      SB    5.6697e-08   W/m^2/K^4 Stefan-Boltzmann constant
3  gasCt1     0.08205784 L*atm/K/mol        ideal gas constant
4  gasCt2       8.314472 m3*Pa/K/mol        ideal gas constant
5     atm         101325          Pa       pressure conversion
6     bar          1e+05          Pa       pressure conversion
7      B1 1.3806504e-23         J/K        Boltzmann constant
8      B2   8.617343e-05        eV/K        Boltzmann constant
```

3. functions

3.1. coriolis

Estimates the coriolis factor, f, units sec^{-1} according to the formula: f=2*omega*sin(lat), where omega=7.292e-5 radians/sec

```
> plot(-90:90,coriolis(-90:90),xlab="latitude, dg North",
+                   ylab= "/s" , main ="coriolis factor",type="l",lwd=2)
```

3.2. heat capacity

Estimates the heat capacity of seawater, using the UNESCO 1983 polynomial (Fofonoff and Millard 1983)

```
> cp(S=40,T=1:20)
```

```
 [1] 3956.080 3955.898 3955.883 3956.021 3956.296 3956.697 3957.209 3957.819
 [9] 3958.516 3959.288 3960.124 3961.013 3961.945 3962.911 3963.900 3964.906
[17] 3965.918 3966.931 3967.936 3968.927
```

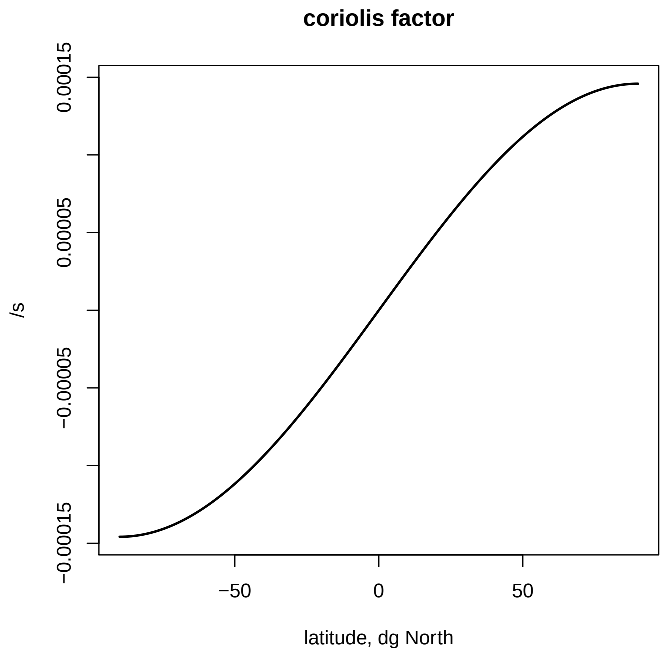

Figure 1: The coriolis function

3.3. molecular diffusion coefficients

Calculates molecular and ionic diffusion coefficients (cm2/hour), for several species at given salinity (S) temperature (T) and pressure (P).

Based on the code "CANDI" by Bernie Boudreau (Boudreau 1996).

```
> diffcoeff(S=15,T=15)*24  # cm2/day
```

```
        O2      CO2       NH3       H2S       CH4      HCO3       CO3      NH4
1 1.429208 1.205458 1.422550 1.229481 1.133012 0.7693272 0.6126977 1.314599
        HS      NO3     H2PO4      HPO4       PO4        H        OH        Ca
1 1.214088 1.283189 0.6168857 0.495435 0.3991121 6.510175 3.543847 0.5264259
         Mg        Fe        Mn      SO4     H3PO4      BOH3      BOH4
1 0.4682133 0.4657005 0.4610938 0.700226 0.5558346 0.7602399 0.6652099
     H4SiO4
1 0.6882129
```

```
> diffcoeff(T=10)$O2
```

```
[1] 0.04930624
```

```
> difftemp <- diffcoeff(T=0:30)[,1:13]
> diffsal <- diffcoeff(S=0:35)[,1:13]
```

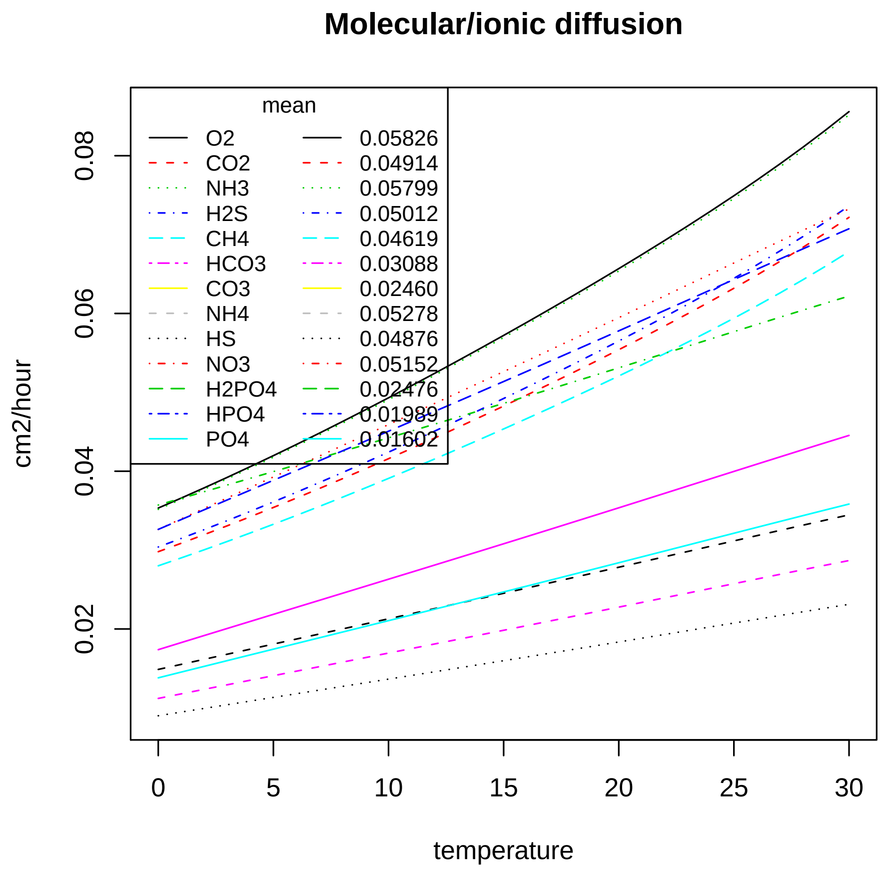

Figure 2: molecular diffusion coefficients as a function of temperature

```
> matplot(0:30,difftemp,xlab="temperature",ylab="cm2/hour",
+         main="Molecular/ionic diffusion",type="l")
> legend("topleft",ncol=2,cex=0.8,title="mean",col=1:13,lty=1:13,
+         legend=cbind(names(difftemp),format(colMeans(difftemp),digits=4)))
```

3.4. molecular diffusion coefficients

Calculates the shear viscosity of water, in centipoise. Valid for 0<T<30 and 0<S<36. Based on the code ”CANDI” by Bernie Boudreau (Boudreau 1996).

```
> plot(0:30,viscosity(S=35,T=0:30,P=1),xlab="temperature",ylab="centipoise",
+        main="shear viscosity of water",type="l")
>   lines(0:30,viscosity(S=0,T=0:30,P=1),col="red")
>   lines(0:30,viscosity(S=35,T=0:30,P=100),col="blue")
>   legend("topright",col=c("black","red","blue"),lty=1,
+         legend=c("S=35,P=1","S=0,P=1","S=35,P=100"))
```

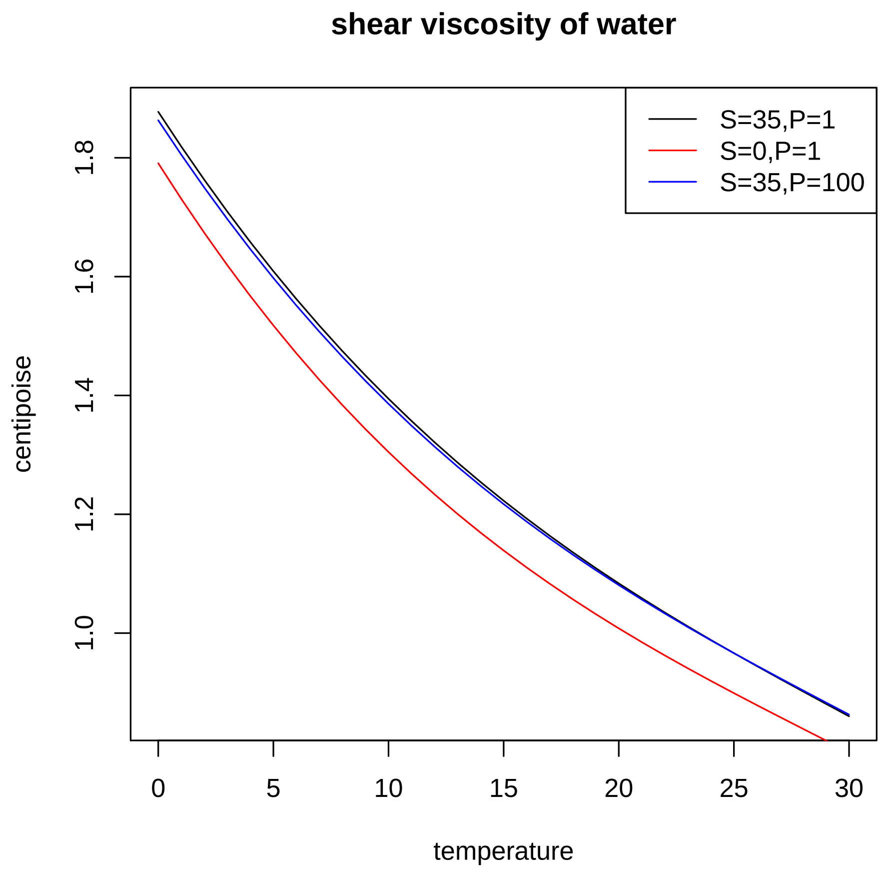

Figure 3: shear viscosity of water as a function of temperature

3.5. Concentration of conservative species in seawater

```
> salconc(S=seq(0,35,by=5))

     Borate   Calcite  Sulphate  Fluoride
1   0.00000     0.000     0.000  0.000000
2  59.42857  1468.571  4033.633  9.760629
3 118.85714  2937.143  8067.267 19.521257
4 178.28571  4405.714 12100.900 29.281886
5 237.71429  5874.286 16134.534 39.042515
6 297.14286  7342.857 20168.167 48.803144
7 356.57143  8811.429 24201.801 58.563772
8 416.00000 10280.000 28235.434 68.324401
```

3.6. Saturated concentration of O2, N2 and Ar

```
> satconc(S=35,T=seq(0,20,by=5))

        O2       N2       Ar
1 358.9267 633.2110 17.44495
2 310.5971 554.2451 15.13654
3 271.9429 490.5970 13.28185
4 240.5663 438.5412 11.76977
5 214.7383 395.3705 10.51985
```

4. conversions

4.1. gram, mol, liter conversions

gram to moles

```
> g2mol("CO3")

       CO3
0.01666419

> g2mol("HCO3")

      HCO3
0.01638892

> g2mol("H")
```

```
        H
0.9921225
```

liter to moles

```
> l2mol(x=8,a=1.382,b=0.03186,T=0)*1000

[1] 357.3925

> l2mol(x=1:6)

[1] 0.04087373 0.08174746 0.12262119 0.16349492 0.20436864 0.24524237
```

molar volume of an ideal gas

```
> mol.vol()

[1] 24.46559

> mol.vol(T=1:10)

 [1] 22.49620 22.57826 22.66032 22.74237 22.82443 22.90649 22.98855 23.07061
 [9] 23.15266 23.23472
```

molecular weight of a chemical species

```
> mol.weight("CO3")

    CO3
60.0089

> mol.weight("HCO3")

    HCO3
61.01684

> mol.weight("H")

      H
1.00794
```

4.2. salinity and chlorinity

```
> sal2cl(S=35)

[1] 19.37394
```

5. finally

This vignette is mainly a Sweave (Leisch 2002) translation of part of the **marelac** help files.

References

Boudreau B (1996). "A method-of-lines code for carbon and nutrient diagenesis in aquatic sediments." *Computers and Geosciences*, **22 (5)**, 479–496.

Fofonoff P, Millard RJ (1983). *Algorithms for computation of fundamental properties of seawater.*, volume 44. Unesco. 53 pp.

Leisch F (2002). "Sweave: Dynamic Generation of Statistical Reports Using Literate Data Analysis." In W Härdle, B Rönz (eds.), "Compstat 2002 - Proceedings in Computational Statistics," pp. 575–580. Physica Verlag, Heidelberg. ISBN 3-7908-1517-9, URL http://www.stat.uni-muenchen.de/~leisch/Sweave.

Soetaert K, Meysman F (2008). *marelac: Datasets, constants, conversion factors, utilities for the marine and lacustrine sciences.* R package version 1.1.

Affiliation:

Karline Soetaert
Centre for Estuarine and Marine Ecology (CEME)
Netherlands Institute of Ecology (NIOO)
4401 NT Yerseke, Netherlands
E-mail: k.soetaert@nioo.knaw.nl
URL: http://www.nioo.knaw.nl/ppages/ksoetaert

Thomas Petzoldt
Institut für Hydrobiologie
Technische Universität Dresden
01062 Dresden, Germany
E-mail: thomas.petzoldt@tu-dresden.de
URL: http://tu-dresden.de/Members/thomas.petzoldt/

Filip Meysman
Laboratory of Analytical and Environmental Chemistry
Vrije Universiteit Brussel (VUB)
Pleinlaan 2
1050 Brussel, Belgium.
E-mail: filip.meysman@vub.ac.be