ES7023 - Assigment 1

Coding Models for Potential Evapotranspiration

The transfer of water from the surface to the atmosphere through evaporation and transpiration (E + T = ET) is a critical process in the Earth System. An accurate picture of ET rates is needed for a range of applications. For one, ET is a major component of soil water balance, so that ET rates influence soil moisture and efforts to predict things like plant production therefore require ET estimates. For another, ET rates influence sensible heating which drives convection. Feel free to learn more about PET by searching on the web or other references (e.g. https://en.wikipedia.org/wiki/Evapotranspiration)

Over the years, people have developed several approaches to modeling potential ET (PET). In this Assignment you will practice coding by creating mathematical funcations for six common methods for modeling PET. You will use these 6 models to calculate PET at a single location. You will also see how there are often many different ways to model the same process, each with different input variables, assumptions and empirical parameters.

To facilitate this work, we have provided some helpful functions for you to use in your models. These are dervided from the FAO methodology (Allen et al. 1998, FAO Irr Drain Paper 56).

Method 1 - Priestly Taylor:

We will write our first PET model together. The Pristely Taylor method for calculating PET is the following:

$$PET = a\frac{s(R_n - G)}{\lambda(s + \gamma)}$$

where:
$a = 1.26$
s = slope of vapor pressure curve
γ = psychrometric constant
G = ground heat flux (assume = 0)
R_n = net radiation
$\lambda = 2.501 - 0.002361(T_{mean})$

In order to `code this in R`, we will write a function for PET, along with several 'helper functions' which will be used *inside* the main PET function. You don't need to worry about the details of the helper functions provided, but feel free to look at their structure to better understand functions in general.

Note that throughout the code provided, I use the "=" assignment operator rather than "<-" as shown in the R-Lab. The use of "<-" is legacy of when R was created (and S before it). Strictly speaking "<-" is a less ambiguous term for "assignment," and is prefered for publishing code, but I am lazy and would rather use a single key "=" than two "<-". So as far as I am concerned it is up to your own preference.

```
#Helper function for net radiation Rn
Rn_fun=function(lat,elev,doy,dew,tmax,tmin,Rs){
  ea=e0_fun(dew)
  Ra=Ra_fun(doy,lat)
  Rso=(0.75+(2e-5)*elev)*Ra          #p85
  Rnl=4.903e-9*(((tmax+273)^4+(tmin+273)^4)/2)*(0.34-0.14*sqrt(ea))*(1.35*(Rs/Rso)-0.35)  #p86
  Rn=(1-0.23)*Rs-Rnl             #(Rns=(1-0.23)Rs) ; Rn=Rns-Rnl p87
  return(Rn)
}
```

```r
#extraterrestrial radiation (MJ / m^2 / day)
Ra_fun=function(doy,lat){   #lat in decimal degrees       #p79 FAO
  psi=(pi/180)*lat
  delta=0.409*sin((2*pi*doy/365)-1.39)
  ws=acos(-tan(psi)*tan(delta))
  dr=1+0.033*cos(2*pi*doy/365)
  Ra=(24*60/pi)*0.082*dr*(ws*sin(psi)*sin(delta)+cos(psi)*cos(delta)*sin(ws))
  return(Ra)
}

#slope of saturation curve
s_fun=function(temp) {4098*(0.6108*exp(17.27*temp/(temp+237.3)))/(temp+237.3)^2}

#psychrometric constant [kPa ?C-1]
psychro_fun=function(elev){             #elev in m
  press=101.3*((293-0.0065*elev)/293)^5.26
  return(0.665e-3*press)
}

#saturated vapor pressure for a given temperature
e0_fun=function(temp) {0.6108*exp(17.27*temp/(temp+237.3))}

tmean_fun=function(tmax,tmin){(tmax+tmin)/2}
```

With these helper functions, we can now write the Pristely Taylor PET function:

```r
PriesTay_PET=function(lat,elev,doy,dew,tmax,tmin,Rs){
  tmean=tmean_fun(tmax,tmin)
  Rn=Rn_fun(lat,elev,doy,dew,tmax,tmin,Rs)
  s=s_fun(tmean)
  y=psychro_fun(elev)
  G=0
  lamda=2.501-0.002361*tmean
  PET=(1.26*((s*(Rn-G))/(s+y)))/lamda
  return(PET)
}

#where:
# lat is latitude
# elev is elevation
# doy is day of year
# dew is dew temperature
# tmax is the daily maximum temperature (in C)
# tmin is the daily min temperature (in C)
# Rs is daily solar radiation
```

We define `PriesTay_PET` by assigning it to the output of function. The list of argument names are contained within parentheses. The body of the function is executed when it is runs, and is contained within curly braces ({}). The statements in the body are indented by two spaces, which makes the code easier to read but does not affect how the code operates.

When we call the function, the values we pass to it are assigned to those variables so that we can use them inside the function. Inside the function, we use a return statement to send a result back to whoever asked for it. The helper functions had the same structure. Note that the `return()` statement is not strictly necesary, as the function will return whichever variable is on the last line of the function, but it is good practice to be explicit.

We can now `call` the function with different input variables as follows:

```
#Run the function Priestay_PET for input values lat=40, elev=50, doy=175, dew=13.9, tmax= 32,
#tmin =11, Rs = 2.5
PriesTay_PET(40,50,175,13.9,32,11,22.5)
```

```
## [1] 4.893226
```

```
#Run the function Priestay_PET for input values lat=42, elev=50, doy=50, dew=13.9, tmax= 32,
#tmin =11, Rs = 2.5
PriesTay_PET(42,50,50,13.9,32,11,22.5)
```

```
## [1] 2.613495
```

You can also run the model for an array of input variables:

```
mult_days=c(50:60)
PriesTay_PET(40,50,mult_days,13.9,32,11,22.5)
```

```
##  [1] 2.847760 2.892801 2.937260 2.981128 3.024395 3.067053 3.109096 3.150519
##  [9] 3.191317 3.231486 3.271025
```

Exercise 1 (10 pts)

Write 5 more functions to compute PET using the five different methods outlined below. The inputs will depend on the model, but the output will always be PET. In all exercises, you can assume that $t_{mean} = \frac{t_{max}+t_{min}}{2}$.

Method 2 - Modified Priestly Taylor:

$$PET = \begin{cases} EEQ * 0.01 * exp(0.18 * (T_{max} + 20)), & T_{max} < 5 \\ EEQ * 1.1, & 5 <= T_{max} <= 24 \\ EEQ * ((T_{max} - 24) * 0.05 + 1.1), & T_{max} > 24 \end{cases}$$

where:
$EEQ = R_s(4.88 \times 10^{-3} - 4.37 \times 10^{-3} * albedo) * (TD + 29)$
$TD = 0.6 * T_{max} + 0.4 * T_{min}$
$albedo = 0.23$
T_{max} = daily maximum temperature (C)
T_{min} = daily minimum temperature (C)

Note that you will likely need to use an `if()` statement for this function, such as:

```
Mod_PriesTay_PET=function(...){
  if(...) {run this code}
}
```

Method 3 - Hamon:

$$PET = 715.5 * DL * \frac{esat}{T_{mean} + 273.2}$$

where:
DL = daylength (days) (from latitude and day of year)
$esat$ = saturated vapor pressure, evaluated at T_{mean}

Note: Function to compute $esat$ and DL are provided here:

```
#this is how FAO suggests one do it for daily esat b/c of nonlinearity of e0 function
esat=function(tmax,tmin) (e0_fun(tmax)+e0_fun(tmin))/2

#daylength (returns # hours in the day with sunlight. only for areas between 65N-6)
daylength=function(lat,doy){
  psi=(pi/180)*lat
  delta=0.409*sin((2*pi*doy/365)-1.39)
  ws=acos(-tan(psi)*tan(delta))
  d=24*ws/pi
  d
}
```

Method 4, Hargreaves:

$$PET = \frac{0.0023 * (T_{mean} + 17.8)(T_{max} - T_{min})^{0.5} * R_a}{\lambda}$$

where:
$\lambda = 2.501 - 0.002361 * T_{mean}$
R_a = extraterrestial radiation (the function for R_a was provided previously)

Method 5 - Linacre:

$$PET = \frac{500 * \dfrac{T_m}{(100 - A)} + 15(T_{mean} - T_{dew})}{80 - T_{mean}}$$

where:
$T_m = T_{mean} + 0.006h$
h = elevation (m)
T_{dew} = dewpoint temp
A = latitude (degrees)

Method 6 - Turc:

$$PET = \begin{cases} (1 + (50 - RH)/70)\dfrac{0.013 * T * (R_s + 50)}{T + 15}, & RH < 50\% \\ \dfrac{0.013 * T * (R_s + 50)}{T + 15}, & RH \geq 50\% \end{cases}$$

where:
T = mean daily temp (C)
R_s = daily solar radiation (units of cal /cm2 which is 23.9 x value in MJ / m^2)
RH = daily mean relative humidity (%)

Exercise 2 (2 pts)

Report PET according to each method for Aug 21, 2013 in Gilroy, CA, which had $Tmax = 27.7C$, $Tmin = 13.3C$, $Tmean = 20.5C$, $RH = 67\%$, $Tdew = 13.9C$, and $R_s = 22.5MJ/m^2$. You will have to look up the coordinates and elevation of Gilroy, CA, and convert the date to "day of year" (in fact there is an R package that can help with that). Compare the results for the various PET models.

Exercise 3 - Start thinking about your project

This exercise is for graduate students only. Undergraduate students who decide to do a project instead of the last assignment must discuss with the teaching team first.

Briefly describe your PhD research topic. Give an example from the literature in your field where models have been used for exploratory/explanatory purposes and for prediction. For your project, you will need to find a dataset relevant to your research, which you can use to investigate a research question of your interest using some of the methods you will learn in the class. Please find at least one suitable dataset in time for the next assignment from e.g. your PhD supervisor, lab colleagues, research collaborators, free online databases or come talk to me if you are interested to use some data from my lab.