Introduction and motivation

The crop water requirement is a key factor in the agricultural process. It is usually estimated throughout actual evapotranspiration ($ET_a$). An accurate quantification of $ET_a$ helps to develop irrigation strategies, improve the efficiency of water use and increase the irrigated area and the production (Millar, 1993; Baruch and Fisher, 1991; Ferreyra et al., 1985).

Traditional methods to estimate $ET_a$ are based on (a) direct measurements by sophisticated instruments, such as lysimeters (Payero and Irmak, 2008; López-Urrea et al., 2009), Eddy covariance systems (Paco et al., 2006; Parent and Anctil, 2012; Poblete-Echeverría and Ortega-Farias, 2013) or Bowen ratios (Crago and Brutsaert, 1996; Ortega-Farias et al., 1995; Twine et al., 2000), or on (b) empirical methods, such as the FAO-56 approach (Allen et al., 1998). This method uses a reference evapotranspiration ($ET_r$) from an automatic weather station multiplied by crop coefficients ($K_c$) from literature (Allen et al., 2005). Although all these methods can be accurate enough, they are restrictive to be extrapolated to a farm or a regional level since they do not take into account the effect that the spatial and temporal variation of the soil, the climate and the crop have over the $ET_a$ (Allen et al., 2011).

However, new physical methods to estimate $ET_a$ have been developed using remote sensing data, considering the land spatial and temporal patterns. A major restriction for the estimation of $ET_a$ using remote sensing is the need of an absolute surface temperature for calibration. One of the first methods to be developed and applied worldwide was the Surface Energy Balance Algorithms for Land (SEBAL) model (Bastiaanssen et al., 1998a,b). This model calculates $ET_a$ from satellite-based land surface energy balance (LSEB) equation and uses a near-surface temperature gradient ($dT$) for calibration. $dT$ is computed by taking two pixels with extreme water condition (anchor pixels) selected in the scene to generate a linear relationship between surface temperature and the difference between surface and air temperatures. Based on this, Allen et al. (2007b) developed the Mapping EvapoTranspiration at High Resolution with Internalized Calibration (METRIC) model. The main difference between SEBAL and METRIC is that the latter uses the $ET_r$ from a weather station, incorporating climatic conditions, while SEBAL uses the potential evaporation from a water body in the scene considering that sensible heat and soil heat fluxes are zero.

METRIC has been widely applied to estimate $ET_a$ at field and regional scale over different crops such as wheat, corn, soybean and alfalfa, with errors ranging between 3 and 20% (Allen et al., 2007a; Choi et al., 2009; Mkhwanazi and Chávez, 2012). In recent years METRIC has been used to compute $ET_a$ over sparse woody canopies such as vineyards and olive orchards (Carrasco-Benavides et al., 2012, 2014; Santos et al., 2012; Pôças et al., 2014) in both flat and mountainous terrains (Allen et al., 2013b).

The current implementations of SEBAL and METRIC imply the need to use more than one software to run the model (Allen et al., 2010) and they involve multiple steps. There are many different sub-models published for the estimation of some parameters (e.g. leaf area index, momentum roughness length, land surface temperature) that are not integrated into the current implementation of METRIC. Allen et al. (2013a) proposed a methodology for an automation procedure by using statistical conditions.

Abstract

The crop water requirement is a key factor in the agricultural process. It is usually estimated throughout actual evapotranspiration ($ET_a$). This parameter is the key to develop irrigation strategies, to improve water use efficiency and to understand hydrological, climatic, and ecosystem processes. Currently, it is calculated with classical methods, which are difficult to extrapolate, or with land surface energy balance models (LSEB), such as METRIC and SEBAL, which are based on remote sensing data. This paper describes water, an open implementation of LSEB. The package provides several functions to estimate the parameters of the LSEB equation from satellite data and proposes a new object class to handle weather station data. One of the critical steps in METRIC is the selection of “cold” and “hot” pixels, which water solves with an automatic method. The water package can process a batch of satellite images and integrates most of the already published sub-models for METRIC. Although water implements METRIC, it will be expandable to SEBAL and others in the near future. Finally, two different procedures are demonstrated using data that is included in water package.
and expert knowledge. This technique reduced the effect of human criteria helping to increase the model robustness. However, a software tool for the automatic selection of anchor pixels has not been published yet.

About land surface energy balance models and crop evapotranspiration

As mentioned above, METRIC estimates \( ET_a \) as the residual from the surface energy balance equation considering information from satellite images and weather stations located near to the study site. Below, the key equations are detailed, beginning with the estimation of \( ET_a \) as the residual from the surface energy balance equation:

\[
LE = R_n - G - H
\]

where \( LE \) is latent heat flux consumed by \( ET_a \) (\( W \cdot m^{-2} \)); \( R_n \) is net radiation (\( W \cdot m^{-2} \)); \( G \) is soil heat flux (\( W \cdot m^{-2} \)); and \( H \) is the sensible heat flux convected to the air (\( W \cdot m^{-2} \)).

\( R_n \) is calculated considering information obtained at the time of satellite overpass. Some correction processes are necessary, such as radiometric and atmospheric corrections. \( G \) is estimated using an empirical equation that considers mainly \( R_n \), surface temperature, normalized difference vegetation index (NDVI), soil-adjusted vegetation index (SAVI) and albedo. More detailed information concerning the equations and models used in METRIC can be found in Allen et al. (2007b).

\( H \) is the general equation of heat transport and is estimated using an approach called “calibration using inverse modeling at extreme conditions” (CIMEC) (Allen et al., 2013a). This method involves the selection of pixels with near extreme conditions (hot and cold anchor pixels) from which the \( ET_a \) can be estimated and assigned. \( H \) is computed as follows:

\[
H = \frac{\rho \cdot c_p \cdot dT}{r_{ah}}
\]

where \( dT \) is the difference between land surface and near-surface air temperatures, \( r_{ah} \) is the aerodynamic resistance to heat transport (\( s \cdot m^{-1} \)), \( \rho \) is the air density (\( kg \cdot m^{-3} \)) and \( c_p \) is the specific heat of air (\( 1004 \cdot J \cdot kg^{-1} \cdot ^{\circ}K^{-1} \)). \( dT \) is solved by using a linear relationship between air temperature and the estimated surface temperature of the anchor pixels (Bastiaanssen et al., 1998a). To calculate \( r_{ah} \), wind speed is extrapolated to a height at which forces of buoyancy and mechanical mix are equal (about 200 meters), using an iterative correction process based on the Monin-Obhukov equations (Allen, 1996; Bastiaanssen et al., 1998a).

After \( LE \) from Equation 1, it is possible to compute the instantaneous evapotranspiration values:

\[
ET_{inst} = 3600 \cdot \frac{LE}{\lambda_{fw}}
\]

where \( ET_{inst} \) is the instantaneous \( ET_a \) at the satellite overpass (\( mm \cdot h^{-1} \)); 3600 is the conversion factor from seconds to hours; \( \rho_{fw} \) is the density of water (\( 1000 kg \cdot m^{-3} \)); and \( \lambda \) is the water latent heat of vaporization (\( J \cdot kg^{-1} \)).

Finally, the daily ET is computed pixel by pixel as:

\[
ET_{24} = \frac{ET_{inst}}{ET_r} \cdot ET_{r,24}
\]

where \( ET_{inst} \) in the instantaneous \( ET_a \) estimated on equation 3; \( ET_r \) is the standardized 0.5 m tall alfalfa reference evapotranspiration at the image time and \( ET_{r,24} \) is the cumulative 24 h ET from the image day. The relationship between \( ET_{inst} \) and \( ET_r \) is the reference ET fraction and is the same as the alfalfa based coefficient, \( K_r \), and is used to extrapolate \( ET_a \) from the image time to periods of 24 hours or longer (Allen et al., 2007b).

Sensible heat flux

As Allen et al. (2007b) mentioned, the computation of latent heat flux \( (LE) \) is only as accurate as the summed estimates for \( R_n \), \( G \), and \( H \). Table 1 shows some errors reported by METRIC models for different crops. It can be seen that net radiation \( (R_n) \) and soil heat flux \( (G) \) present the lowest
estimation errors, while sensible heat flux ($H$), is the hardest component of the surface energy balance to estimate.

One of the weaknesses of METRIC model reported in the literature is the selection of the anchor pixels. Long and Singh (2013) and Morton (2013) indicated that the selection of anchor pixels is subjective and depends on the ability of the operator to search and isolate the most appropriate hot and cold pixels. This process produces important biases in the estimation of $H$. Also, Choragudi (2011); Wang et al. (2009) mentioned that METRIC was very sensitive to the selection of the hot pixel. A group of possible candidates could have minimal differences in some attributes, but these can generate a big bias in the estimations. It means that the estimation of $H$ in METRIC is very sensitive to the selection of anchor pixels.

The objective of this article is to propose an open implementation of a land surface energy balance model (LSEB) as an R package that integrates most of the METRIC sub-models and allows automatic selection of the anchor pixels. In this version of the package, specific functions for METRIC model (Allen et al., 2007b) are provided. Apart from the previous features, this package: i) is written in R, one of the most used scientific programming languages; ii) can be automated for batch processing of many satellite images; iii) provides functions for loading and processing satellite images; iv) provides functions and a new class object to manage weather station data; and v) is a free and open software.

### About the water package

#### Package organization

The water package is developed in R to estimate actual evapotranspiration ($ET_a$) from Landsat satellite scenes using Land Surface Energy Balance (LSEB) models, such as METRIC (Allen et al., 2007b).

Functions in water package are arranged in three groups: i) general functions to estimate sub-components of LSEB (e.g. leaf area index, albedo, land surface temperature, momentum roughness length); ii) specific functions to estimate the components of LSEB; iii) internal functions and methods to handle data from a weather station using a new proposed S3 class and functions to control global options such as saving results to disk, overwrite files, etc.

The first group of functions consist of models and equations to estimate the sub-components, which generally are controlled by the argument method. Most of the models available here are presented in the Appendix. In the second group, there are three functions: i) METRIC.Rn(); ii) METRIC.G() and iii) METRIC.ED(). The first one estimates net radiation using METRIC model. The second one estimates soil heat flux, and the third one estimate all the components of energy balance: $R_n$, $G$, $H$ and $LE$ according to METRIC model.

Three example datasets are provided with the water package: i) a subset of a Landsat 7 scene path 223, row 85 from 15th February 2013, bands 1 to 7; ii) data from a weather station from the same Landsat subset in CSV file format (comma separated values); iii) a subset of NASA SRTM digital elevation model, with the same spatial extent as the example image. These datasets are used in examples in Sections Simple procedure and Advanced procedure, and also in the vignettes included with the package.

The water package is available on The Comprehensive R Archive Network at https://CRAN.R-project.org/package=water. This software is made freely available under the terms and conditions of the GNU General Public License.

### Key functions in the water package

The key functions included in the package are:

<table>
<thead>
<tr>
<th>Crop</th>
<th>Validation tool</th>
<th>$R_n$ ($Wm^{-2}$%)</th>
<th>$G$ ($Wm^{-2}$%)</th>
<th>$H$ ($Wm^{-2}$%)</th>
<th>$LE$ ($Wm^{-2}$%)</th>
<th>$ET$ (mm/1%)</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>grass</td>
<td>lysimeter</td>
<td>nr</td>
<td>nr</td>
<td>nr</td>
<td>nr</td>
<td>(4-22%)</td>
<td>(4%) 1</td>
</tr>
<tr>
<td>sugar beet</td>
<td>lysimeter</td>
<td>nr</td>
<td>nr</td>
<td>nr</td>
<td>nr</td>
<td>(6-13%)</td>
<td>(1%) 2</td>
</tr>
<tr>
<td>corn, soybean</td>
<td>EC</td>
<td>nr</td>
<td>nr</td>
<td>nr</td>
<td>nr</td>
<td>0.14</td>
<td>(17.6%) 3</td>
</tr>
<tr>
<td>olive</td>
<td>EC</td>
<td>nr</td>
<td>nr</td>
<td>nr</td>
<td>nr</td>
<td>0.14-1.2</td>
<td>4</td>
</tr>
<tr>
<td>vineyard</td>
<td>EC</td>
<td>24 (3.8%)</td>
<td>16 (9.4%)</td>
<td>39-59 (10-26.0%)</td>
<td>33-54 (14-27.2%)</td>
<td>0.9 (9%)</td>
<td>5</td>
</tr>
</tbody>
</table>

nr: not reported; EC: Eddy covariance
Sources: 1: Allen et al. (2007a); 2: Mkhwanazi and Chávez (2012, 2013); 3: Gonzalez-Dugo et al. (2009); 4: Santos et al. (2012); Pôças et al. (2014); 5: Poblete-Echeverría and Ortega-Farias (2012); Carrasco-Benavides et al. (2012, 2014)
This function allows to import weather station data from a table in comma-separate values (CSV) format. The result is a new object of class "waterWeatherStation". The main input arguments are the CSV file and a vector with the order of the needed variables (radiation, temperature, wind speed and relative humidity) called columns. An optional argument is a Landsat metadata file (MTL). When this function is used with the CSV and the MTL files, it will interpolate the weather conditions to the moment of the satellite overpass.

\textbf{METRIC.EB()} This is the main function of \texttt{water} package. It runs each of the sub-models needed to get all the components of the LSEB (equation 1), from satellite and weather station data. The input arguments are: a satellite scene and a "waterWeatherStation" object. The arguments \texttt{alb.coeff}, \texttt{LAI.method}, \texttt{Zom.method} and \texttt{anchors.method} allow choosing between the different sub-models or coefficients used. More information about the sub-models available in \texttt{water} is presented in the Appendix. An optional logical argument is \texttt{plain}, which allows to use a digital elevation model or to consider that the surface is flat. When using a digital elevation model, the net radiation estimation and the land surface temperature are corrected using the elevation, slope and aspect of the surface. The output of this function is a raster layer object (from \texttt{raster} package) with 4 different layers: net radiation, soil heat flux, sensible heat flux and surface temperature.

\textbf{ET24h()} This function estimates the $ET_r$ for a 24-hour period. The input arguments are the components of the LSEB ($R_n, G, H$) and a "waterWeatherStation" object. The argument \texttt{ET} allows to select between two $ET_r$ methods. $ET_r$ is the method for short crops, similar to clipped, cool-season grass and $ET_{o}$, is the method for tall crops, similar to 0.5 m tall full-cover alfalfa. By default, this method will use $ET_r$ and the user should choose according to the conditions of the weather station. The output of this function is a raster layer object (from \texttt{raster} package) with the 24-hour $ET_r$ in mm·day$^{-1}$.

\textbf{METRIC.EB()} uses many different steps to estimate the parameters needed to calculate the components of the LSEB. These steps are available as individual functions like:

\textbf{albedo()} This function is used to calculate the broadband albedo from narrowband satellite data. This process involves applying a weighting function with empirical coefficients. \texttt{water} includes models and coefficients described by Tasumi et al. (2008) and Liang (2001), as well as new coefficients for Landsat 8 estimated by The Simple Model of the Atmospheric Radiative Transfer of Sunshine (SMARTS2) version 2.9.5 (Gueymard, 1995). \texttt{coeff="Olmedo"} computes clear sky spectral irradiances for the spectral range of each Landsat 8 OLI band (see Appendix, Section Included models for albedo). The output of this function is a raster layer object (from \texttt{raster} package) with the albedo.

\textbf{LAI()} This function estimates the leaf area index (LAI) from Landsat data. \texttt{water} includes many different models to estimate LAI (see Appendix, Section Included models for Leaf Area Index). In the Examples Section of this article, the METRIC 2010 method (Allen et al., 2010) will be used. This method estimates LAI from SAVI (Huete, 1988), as follows:

\begin{equation}
SAVI = (1 + L)(\rho_{NIR} - \rho_R)/(L + \rho_{NIR} + \rho_R)
\end{equation}

where $\rho$ is the reflectance at the top-of-atmosphere from $R$, the red band, and from $NIR$, the near infrared band; $L$ is a soil correction factor. By default \texttt{water} uses $L=0.5$, although Allen et al. (2007b) suggested a value of $L=0.1$ in METRIC applications for western USA. This value varies by the amount or coverage of green vegetation: in very high coverage vegetation regions, $L=0$ and $SAVI = NDVI$; in areas with no green vegetation, $L=1$. Then SAVI is used to estimate LAI as follows:

\begin{equation}
LAI = 11 \cdot SAVI^3
\end{equation}

The output of this function is a raster layer object (from \texttt{raster} package) with the estimated $LAI$.

As it was mentioned before, the critical part in the estimation of the LSEB is the sensible heat flux estimation. The key functions used to estimate this are: \texttt{momentumRoughnessLength()}, \texttt{calcAnchors()} and \texttt{calcH()}.  

\textbf{momentumRoughnessLength()} This function estimates the Momentum Roughness Length ($Z_{om}$) from the average vegetation height around the weather station. \texttt{water} includes several methods to estimate $Z_{om}$ from Landsat data (see Appendix, Section Included models for Momentum Roughness Length). For example, when \texttt{method="short-crops"} (Allen et al., 2007b) $Z_{om}$ is estimated as:
Variable | Cold pixel | Hot pixel  
--- | --- | ---  
albedo* | 0.18 - 0.25 | 0.13 - 0.15  
NDVI* | 0.76 - 0.84 | 0.10 - 0.28  
LAI (m$^2$ · m$^{-2}$) | 3 - 6 | -  
$Z_{om}$ (m) | 0.03 - 0.08 | ≤ 0.005  

* dimensionless

Table 2: Ranges of variables used for selection of anchor (cold and hot) pixels in calcAnchors(method="CITRA-MCB")

$$Z_{om} = 0.0018 \text{LAI}$$

(7)

And if mountainous=TRUE, this value of $Z_{om}$ is corrected as:

$$Z_{om,\text{mtn}} = Z_{om} \cdot (1 + (\frac{180}{\pi} \cdot \text{slope} - 5) \cdot \frac{20}{20})$$

(8)

The output of this function is a raster layer object with the estimated $Z_{om}$.

calcAnchors() This function automatically select anchor pixels that represent the dry and wet ends of the ET spectrum within the satellite scene using information of land surface characteristics (LAI, albedo, $Z_{om}$, $T_s$). When the anchor pixels are found, it assigns the $ET_a$ estimates.

The criteria to select “hot” and “cold” pixels were adapted from Allen et al. (2011), Tasumi (2003) and shown in Table 2. The methodology searches for image-specific pixels with the lowest and highest temperature values that match with these criteria. The output of this function is a data frame with the coordinates of the selected anchor pixels.

calcH() This function applies the CIMEC self-calibration method in order to generate an iterative process for the “hot” and “cold” pixels and absorb all biases in the computation of $H$. A near surface temperature difference ($dT$) is used in place of a surface air temperature difference to drive the determination of sensible heat flux, in an iterative correction process. The convergence of the function is reached when the change in the estimated aerodynamic resistance is less than 1% for the cold and hot condition. There is an argument called verbose to control how much information about the iterative process is shown in the output. The output of this function is a raster layer object with the estimated soil heat flux.

Input data requirements

METRIC uses two sources to estimate the LSEB: a satellite image and a weather station with hourly data.

METRIC can be run using different satellite sensors. Currently, the coefficients needed for Landsat 7 and 5 are available in Allen et al. (2007b). Those coefficients can also be applied to Landsat 8 data. In the Appendix Section Included models for albedo, we propose specific coefficients for the estimation of albedo using this satellite sensor. Other coefficients to run METRIC using MODIS images are available in Allen et al. (2007b), and will be included in future versions of water package.

The weather station data should include near-surface air temperature, wind speed, relative humidity and solar radiation. The water package uses the function read.WSdata() to convert the weather station data from a comma-separate values table to a special R object. The input data should be on an hourly or shorter time basis. The expected units are °C for temperature; m · s$^{-1}$ for wind speed; % for relative humidity and W · m$^{-2}$ for solar radiation. However, this function uses a parameter cf when conversion factors are needed to convert the variables from different units (e.g. if wind speed is in km · h$^{-1}$ the conversion factor for this variable should be 0.278). To estimate $ET_r$ from the weather station data using Allen et al. (2005) equation, more information is needed: position in latitude and longitude, and the wind sensor height in meters. When the weather station data is being imported, a satellite metadata file can be included as a function argument. This allows interpolating the weather conditions at the exact moment of the satellite overpass.

The water package only uses one weather station to estimate $ET_r$ for an entire Landsat scene of 180 × 180 km. The model uses $ET_r$ to derive the ET reference fraction ($ET_rF$) at image time (using equation 4). This assumes that $ET_a$ in the entire area changes in proportion to the change in $ET_r$ at the weather station (Allen et al., 2007b). This means that $ET_r$ is only used as an index of the relative
Figure 1: Plot of the hourly weather station data. The conditions at the time of the satellite overpass are marked by a gray bar. This is the default plot for `waterWeatherStation` objects.

Change and this is retained through the $ET_r$. Any biases caused by variation in weather conditions should be canceled by using the same $ET_r$ for both instantaneous and 24 h period. Nevertheless, it is recommended to use a spatial mask when the weather conditions are heterogeneous, for example in irrigated areas surrounded by deserts.

Performance and memory use

The `water` package uses large temporal memory in order to obtain the results and sub products. Most of the results are "RasterLayer" or "RasterStack" objects from `raster` package (Hijmans, 2015). Processing an entire Landsat scene could need more than 2 gigabytes of memory to store the temporal data. One approach to solve this is to write products and sub-products to the disk. There is an option `writeResults=TRUE` in function `waterOptions()` to force `water` to store the results on the disk, instead of on temporal memory.

If `water` runs out of memory while processing data, it will usually stop working without a warning message. We suggest processing only a portion of a Landsat scene using an area-of-interest (aoi) polygon or storing results to disk.

Example code and datasets

Two different approaches to estimate land surface energy balance demonstrate the features and procedures in `water`. The first example in Section Simple procedure is a simple procedure, and the second one in Section Advanced procedure refers to an advanced procedure. Finally, the estimation of $ET_a$ from the output of any of the previous procedures is demonstrated in Section Daily crop evapotranspiration estimation.

In Section Estimating ETa using METRIC model and water package functions `createAoi()` and `read.WSdata()` are summarized. Then, in Section Simple procedure function `METRIC.EB()` is shown. In Section Advanced procedure the LSEB is estimated step by step. And later, in Section Daily crop evapotranspiration estimation daily $ET_r$ is estimated using functions `dailyET()` and `ET24h()`.

Estimating ETa using METRIC model and water package

Base data preparation

To calculate METRIC Actual Evapotranspiration using `water` package, three sources are needed:

1. A raw Landsat 7-8 satellite image.
2. A Weather Station data (.csv file).
3. A polygon with our Area-of-interest (AOI) Spatial-Polygon object, to run the model using only a portion of the satellite scene.

First, AOI is created as a polygon using bottomright and topleft coordinates:

```r
aoi <- createAoi(topleft = c(272955, 6085705),
                 bottomright = c(288195, 6073195), EPSG = 32719)
```

Base data preparation

To calculate METRIC Actual Evapotranspiration using `water` package, three sources are needed:

1. A raw Landsat 7-8 satellite image.
2. A Weather Station data (.csv file).
3. A polygon with our Area-of-interest (AOI) Spatial-Polygon object, to run the model using only a portion of the satellite scene.

First, AOI is created as a polygon using bottomright and topleft coordinates:

```r
aoi <- createAoi(topleft = c(272955, 6085705),
                 bottomright = c(288195, 6073195), EPSG = 32719)
```
Figure 2: Schematic diagram with the functions, data inputs and outputs used in the simple procedure when running the METRIC model with \texttt{water} package. The green rounded boxes represent data, the blue boxes represent the functions and the yellow box is the final result.

Then, the weather station data is loaded using the function \texttt{read.WSdata()}. This function converts the CSV file into a "waterWeatherStation" object. Then, if a Landsat metadata file (MTL file) is provided, the time-specific weather conditions at the time of satellite overpass will be calculated. This is shown on Figure 1 as a gray bar. Files ‘apples.csv’ and ‘L7.MTL.txt’ are included in the package as raw data. In R, \texttt{system.file()} is used to call this files.

\begin{verbatim}
csvfile <- system.file("extdata", "apples.csv", package = "water")
MTLfile <- system.file("extdata", "L7.MTL.txt", package = "water")
WeatherStation <- read.WSdata(WSdata=csvfile,
    date.format = "%d/%m/%Y",
    lat = -35.42222, long = -71.38639,
    elev = 201, height = 2.2,
    MTL = MTLfile)
\end{verbatim}

Next, the Landsat satellite image is loaded. \texttt{water} provides a function to load a Landsat image (\texttt{loadImage()}) from TIFF files. Landsat images can be downloaded directly from USGS archives in \texttt{Earth Explorer} (http://earthexplorer.usgs.gov/). In this article, an example dataset will be used which comes with \texttt{water} package as demonstration data.

\begin{verbatim}
image.DN <- L7_Talca
\end{verbatim}

Finally, Digital Elevation Model (DEM) will be created for the area being processed. \texttt{water} provides two functions to do this: \texttt{checkSRTMgrids()} will search for the downloadable grid files in http://earthexplorer.usgs.gov/. However, this function will only print the links to the files. The downloading process has to be done manually. After this, \texttt{prepareSRTMdata()} can be used to mosaic and clip those files using the same extent of the image. In this article, the example data, provided with \texttt{water} package, will be loaded.

\begin{verbatim}
DEM <- DEM_Talca
\end{verbatim}

Simple procedure

The simple procedure is summarized on Figure 2. The function \texttt{METRIC.EB()} will be used to estimate the land surface energy balance. This function has many parameters to choose from the different METRIC model equations. e.g. changes can be made in:

- Coefficients used to estimate broadband albedo from narrowband data.
- Model to estimate Leaf Area Index (LAI) from satellite data.
Figure 3: Land surface energy balance \((W \cdot m^{-2})\) estimated using METRIC with \texttt{water} package

- Model to estimate momentum roughness length \((Z_{om})\)
- Automatic method for the selection of anchor pixels
- Reference ET coefficient and momentum roughness length estimated for the weather station

When this function is run, the energy balance and the surface temperature \((T_s)\) used are assigned to the \texttt{Energy.Balance} object. This function prints the position and characteristics of the anchor pixels to the console. Also, a plot with the values of the aerodynamic resistance during the iterative process is generated after every iteration. Here, the logical argument \texttt{verbose} controls how much information is shown in the output, and the plotting of the diagnostic graph.

\[
\text{Energy.Balance} \leftarrow \text{METRIC.EB(image.DN = image.DN,}
\right.
\]
\[
\left. \text{plain = FALSE, DEM = DEM,}
\right.
\]
\[
\left. \text{WeatherStation = WeatherStation, ETp.coef = 1.2,}
\right.
\]
\[
\left. \text{MTL = MTLfile, sat = "L7",}
\right.
\]
\[
\left. \text{thermalband = image.DN$thermal.low)\right.\]
\]

The results of the energy balance estimated using this function are shown in Figure 3. The console output with information related to the anchor pixels goes like this:

<table>
<thead>
<tr>
<th>pixel</th>
<th>X</th>
<th>Y</th>
<th>Ts</th>
<th>LAI</th>
<th>type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>139253</td>
<td>282420</td>
<td>-3922830</td>
<td>323.1587</td>
<td>0.13</td>
</tr>
<tr>
<td>2</td>
<td>121566</td>
<td>274710</td>
<td>-3921780</td>
<td>310.0151</td>
<td>4.40</td>
</tr>
</tbody>
</table>

Advanced procedure

The advanced procedure involves running many different functions one-by-one, this is summarized in figures 4 and 5. These functions were run inside the code of \texttt{METRIC.EB()} in the previous example. Running \texttt{water} with this procedure allows to have more control in the different arguments used.

Net Radiation estimation

In order to calculate the \(R_n\) for the loaded Landsat satellite data, a surface model (slope + aspect) from the DEM is calculated, then the solar angles (latitude, declination, hour angle and solar incidence angle) are calculated. Then \texttt{incSWradiation()} is used to calculate incoming solar radiation.

\[
surface.model \leftarrow \text{METRICtopo(DEM)}
\]
\[
solar.angles.r \leftarrow \text{solarAngles(surface.model = surface.model,}
\right.
\]
\[
\left. \text{WeatherStation = WeatherStation, MTL = MTLfile)\right.\]
\]
\[
Rs.inc \leftarrow \text{incSWradiation(surface.model = surface.model,}
\right.
\]
\[
\left. \text{solar.angles = solar.angles.r,}
\right.
\]
\[
\left. \text{WeatherStation = WeatherStation)\right.\]
\]
Figure 4: Schematic diagram with the functions, data inputs and outputs used in the advanced procedure when estimating Net Radiation and Soil Heat Flux running the METRIC model with \texttt{water} package. The green rounded boxes represent data and the blue boxes represent the functions. The functions marked with "*" have multiple methods available.
After this, reflectances are calculated at the top-of-atmosphere (TOA), and surface reflectance derived from the Landsat image as:

```r
image.TOAr <- calcTOAr(image.DN = image.DN, sat = "L7", MTL = MTLfile,
                       incidence.rel = solar.angles.r\$incidence.rel)
```

```r
image.SR <- calcSR(image.TOAr = image.TOAr, sat = "L7",
                    surface.model = surface.model,
                    incidence.hor = solar.angles.r\$incidence.hor,
                    WeatherStation = WeatherStation, ESPA = FALSE)
```

Following this, **broadband albedo** is calculated as the sum of visible to near infrared narrowband satellite bands and coefficients related to atmospheric transmittance of global solar beam radiation. In this example `coeff="Tasumi"` was used.

```r
albedo <- albedo(image.SR=image.SR, coeff="Tasumi")
```

Later on, **Leaf Area Index (LAI)** is calculated using the satellite data. In this example `method=metric2010` is used:

```r
LAI <- LAI(method="metric2010", image=image.TOAr, L=0.1)
```

Land surface temperature ($T_s$) is estimated using computed LAI values in order to estimate consequently the surface emissivity and brightness temperature from Landsat’s thermal band (TIR). Then this information is used to compute the incoming and outgoing long-wave radiation as:

```r
Ts <- surfaceTemperature(LAI = LAI, sat = "L7",
                         thermalband = image.DN\$thermal.low,
                         WeatherStation = WeatherStation)
```

```r
Rl.out <- outLWradiation(LAI = LAI, Ts = Ts)
```

```r
Rl.inc <- incLWradiation(WeatherStation, DEM = surface.model\$DEM,
                          solar.angles = solar.angles.r, Ts = Ts)
```

Finally, **Net Radiation ($R_n$)** can be estimated pixel by pixel as follows:

```r
Rn <- netRadiation(LAI, albedo, Rs.inc, Rl.inc, Rl.out)
```

**Soil Heat Flux estimation**

Soil heat flux is estimated $G$, using as input data the $R_n$, surface reflectance, $T_s$, LAI and albedo. In this example the original METRIC (2007) based-method will be used, which is:

```r
G <- soilHeatFlux(image = image.SR, Ts = Ts, albedo = albedo,
                  Rs = Rn, LAI = LAI)
```

**Sensible Heat Flux estimation**

To estimate the sensible heat fluxes derived from the Landsat satellite data, first, the calculation of the momentum roughness length ($Z_{om}$) is needed.

```r
Z.om <- momentumRoughnessLength(LAI = LAI, mountainous = TRUE,
                                 method = "short.crops",
                                 surface.model = surface.model)
```

Then, `calcAnchors()` is used to search for the anchor pixels within the Landsat scene. And finally, `calcH()` is used to run the CIMEC process and estimate the sensible heat flux:

```r
hot.and.cold <- calcAnchors(image = image.TOAr, Ts = Ts,
                             LAI = LAI, plots = FALSE, albedo = albedo,
                             Z.om = Z.om, n = 1,
                             anchors.method = "CITRA-MCB",
                             deltaTemp = 5, verbose = FALSE)
```

```r
H <- calcH(anchors = hot.and.cold, Ts = Ts, Z.om = Z.om,
            WeatherStation = WeatherStation, ETp.coef = 1.05,
            Z.om.ws = 0.0018, DEM = DEM, Rn = Rn, G = G, verbose = TRUE)
```
Figure 5: Schematic diagram with the functions, data inputs and outputs used in the advanced procedure when estimating the sensible heat flux running the METRIC model with water package. The green rounded boxes represent data, the blue boxes represent the functions and the yellow box is the final result. The functions marked with * have multiple methods available.

When the function calcH() is running, and verbose=TRUE, the output shows the intermediate values of the CIMEC process parameters. Also, the value for the aerodynamic resistance and its change for every iteration is plotted iteratively by this function. This plot is shown on Figure 6. In this example, the change in the value of the aerodynamic resistance goes down after iteration #9, and in iteration #14 is less than 1%.

Figure 6: Leaf area index \( m^2 m^{-2} \) and the position of the hot pixel (red cross), cold pixel (blue X) and the weather station (circle with a X) (left). Change on aerodynamic resistance convergence in the iterative process for hot (red) and cold (blue) conditions (right).

Daily crop evapotranspiration estimation

To estimate the daily actual evapotranspiration from the Landsat scene, the daily reference ET \( (ET_r) \) is needed. The daily \( ET_r \) can be calculated with dailyET(). This function calculates the cumulative 24h standardized reference evapotranspiration for the day of the image using the ASCE standardized Penman-Monteith method (Allen et al., 2005). Finally, 24h crop ET can be estimated for every pixel of the Landsat scene using the function ET24h() (Figure 7):

\[
ET_{WS} \leftarrow \text{dailyET}(\text{WeatherStation} = \text{WeatherStation}, \ ET = "ETr")
\]

\[
\text{ET.24} \leftarrow \text{ET24h}(Rn = \text{Rn}, \ G = \text{G}, \ H = \text{H}$\text{H}, \ Ts = \text{Ts}, \ \text{WeatherStation} = \text{WeatherStation}, \ ETr.daily = ET_{WS})
\]
Conclusions

The water package offers a fast and reliable platform for estimating actual evapotranspiration using the land surface energy balance. It includes different methods for many sub-models. The simple procedure showed in this article allows to estimate the ET in a simple and fast way. The advanced procedure allows to have more control in the different available methods. Further versions of this package will implement other LSEB models such as SEBAL. Also, other satellite sensors such as MODIS will be included. Because water is written in R, a language very popular in the scientific community and published as free software, further developments could come from a wide community of users.

Acknowledgments

This work was supported by the Argentinian government through the projects INTA 1133043 and 1133042, and Universidad de Talca, Chile through the research program “Adaptation of Agriculture to Climate Change (A2C2)”. The authors wish to thank Danlu Guo and an anonymous reviewer for their careful reading of the manuscript and their enriching comments. Finally, we thank Marcos Angelini, Rosana Vallone and Marcos Carrasco-Benavides for their valuable support and suggestions to improve this work.

Bibliography


J. C. Jimenez-Munoz, J. Cristobal, J. A. Sobrino, G. S??ria, M. Ninyerola, and X. Pons. Revision of
the single-channel algorithm for land surface temperature retrieval from Landsat thermal-infrared
10.1109/TGRS.2008.2007125. [p368]

J. C. Jimenez-Munoz, J. A. Sobrino, D. Skokovic, C. Mattar, and J. Cristobal. Land surface temperature
retrieval methods from Landsat-8 thermal infrared sensor data. *IEEE Geoscience and Remote Sensing

ISSN 01681699. doi: 10.1016/S0168-1699(02)00106-0. [p367]


D. Long and V. P. Singh. Assessing the impact of end-member selection on the accuracy of satellite-
based spatial variability models for actual evapotranspiration estimation. *Water Resources Research*,

R. López-Urrea, a. Montoro, P. López-Fuster, and E. Fereres. Evapotranspiration and responses to
irrigation of broccoli. *Agricultural Water Management*, 96(7):1155–1161, 2009. ISSN 03783774. doi:
10.1016/j.agwat.2009.03.011. [p352]


M. M. Mkhwanazi and J. L. Chávez. Using METRIC to estimate surface energy fluxes over an alfalfa
State University, 2012. [p352, 354]

M. M. Mkhwanazi and J. L. Chávez. Mapping evapotranspiration with the remote sensing ET algo-
rithms METRIC and SEBAL under advective and non-advective conditions: accuracy determination
with weighing lysimeters. (970), 2013. [p354]

C. G. Morton. **Assessing Calibration Uncertainty and Automation for Estimating Evapotranspiration from
Agricultural Areas Using METRIC.** PhD thesis, Department of Geography, University of Nevada,
2013. [p354]

S. Ortega-Farías, R. Cuenca, and M. English. Hourly grass evapotranspiration in modified maritime

A.-C. Parent and F. Anctil. Quantifying evapotranspiration of a rainfed potato crop in South-eastern
Canada using eddy covariance techniques. *Agricultural Water Management*, 113:45 – 56, 2012. ISSN
0378-3774. doi: http://dx.doi.org/10.1016/j.agwat.2012.06.014. [p352]

J. O. Payero and S. Irmak. Construction, installation, and performance of two repacked weighing
[p352]

T. Paço, M. Ferreira, and N. Conceição. Peach orchard evapotranspiration in a sandy soil: Comparison
between eddy covariance measurements and estimates by the FAO 56 approach. *Agricultural Water
014. [p352]

C. Poblete-Echeverría and S. Ortega-Farias. Calibration and validation of a remote sensing algorithm
to estimate energy balance components and daily actual evapotranspiration over a drip-irrigated
0381-x. [p354]

C. Poblete-Echeverría and S. Ortega-Farias. Evaluation of single and dual crop coefficients over a
drip-irrigated Merlot vineyard (Vitis vinifera L) using combined measurements of sap flow sensors

Allen. Satellite-based evapotranspiration of a super-intensive olive orchard: Application of METRIC
[p352, 354, 367, 369]


Appendix

Included models for albedo

1. coeff="Tasumi" (Tasumi et al., 2008)

\[
\text{albedo} = \rho_{s,B} \cdot 0.254 + \rho_{s,G} \cdot 0.149 + \rho_{s,R} \cdot 0.147 + \rho_{s,NIR} \cdot 0.311 + \rho_{s,SWIR1} \cdot 0.103 + \rho_{s,SWIR2} \cdot 0.036
\]  

(9)

2. coeff="Liang" (Liang, 2001)

\[
\text{albedo} = \rho_{s,B} \cdot 0.356 + \rho_{s,R} \cdot 0.130 + \rho_{s,NIR} \cdot 0.373 + \rho_{s,SWIR1} \cdot 0.085 + \rho_{s,SWIR2} \cdot 0.072 - 0.0018
\]  

(10)

3. coeff="Olmedo"

\[
\text{albedo} = \rho_{s,B} \cdot 0.246 + \rho_{s,G} \cdot 0.146 + \rho_{s,R} \cdot 0.191 + \rho_{s,NIR} \cdot 0.304 + \rho_{s,SWIR1} \cdot 0.105 + \rho_{s,SWIR2} \cdot 0.008
\]  

(11)

where \( \rho_{s,b} \) is the surface reflectance for band \( b \).

Included models for Leaf Area Index

1. method="metric" (Allen et al., 2007b)

\[
\text{SAVI}_{ID} = \frac{(1 + L)(\rho_{L,NIR} - \rho_{L,R})}{(L + \rho_{L,NIR} + \rho_{L,R})}
\]  

(12)

where \( \rho \) is the reflectance at top-of-atmosphere, and the subindex refers to bands \( R: \text{red} \) or \( NIR: \text{near infrared} \); and \( L \) is a soil correction factor. The default value used for \( L=0.5 \), and in METRIC applications in western US, Allen et al. (2007b) suggested a value of \( L=0.1 \). And Leaf Area Index is:

\[
\text{LAI} = -\ln\left(\frac{(0.69 - \text{SAVI}_{ID})/0.59}{0.91}\right)
\]  

(13)

2. method="metric2010" (Pôças et al., 2014)

\[
\text{LAI} = 11 \cdot \text{SAVI}_{ID}^3
\]  

(14)

3. method="vineyard" (Johnson et al., 2003)

\[
\text{NDVI} = \frac{(\rho_{L,NIR} - \rho_{L,R})}{(\rho_{L,NIR} + \rho_{L,R})}
\]  

(15)

where \( \rho \) is the reflectance at top-of-atmosphere, and the subindex refers to bands \( R: \text{red} \) or \( NIR: \text{near infrared} \). And Leaf Area Index is:

\[
\text{LAI} = 4.9 \cdot \text{NDVI} - 0.46
\]  

(16)

4. method="MCB" (Carrasco-Benavides et al., 2014)

\[
\text{LAI} = 1.2 - 3.08 \cdot \exp(-2013.35 \cdot \text{NDVI}^6.41)
\]  

(17)

5. method="turner" (Turner et al., 1999)

\[
\text{NDVI} = \frac{(\rho_{L,NIR} - \rho_{L,R})}{(\rho_{L,NIR} + \rho_{L,R})}
\]  

(18)
where $\rho$ is the reflectance at surface level, and the subindex refer to bands $R$: red or $NIR$: near infrared. And Leaf Area Index is:

$$\text{LAI} = 0.5724 + 0.0989 \cdot \text{NDVI} - 0.0114 \cdot \text{NDVI}^2 + 0.0004 \cdot \text{NDVI}^3$$  \hspace{1cm} (19)

**Included models for Land Surface Temperature**

1. **method="metric"** (Allen et al., 2007b) (Landsat 7)

$$\epsilon_{NB} = 0.97 + 0.0033 \cdot \text{LAI}$$ \hspace{1cm} (20)

and $\epsilon_{NB} = 0.98$ when $\text{LAI} > 3$; where $\epsilon_0$ is the broadband surface emissivity (dimensionless); and $\text{LAI}$ is the leaf area index ($m^2 \cdot m^{-2}$).

$$T_s = \frac{K_2}{\ln[(\epsilon_{NB}K_1/R_c) + 1]}$$ \hspace{1cm} (21)

where $T_s$ is the land surface temperature ($K$); $K_1$ and $K_2$ are specific constants for Landsat 7 ($K_1 = 666.1$ and $K_2 = 1283 W \cdot m^{-2} \cdot sr^{-1} \cdot \mu m$); and $R_c$ is the corrected thermal radiance from the surface using the spectral radiance from band 6 of Landsat following Wukelic et al. (1989).

2. **method="SC"** (Jimenez-Munoz et al., 2009, 2014) (Landsat 8)

$$T_s = \gamma \left[ \frac{1}{\epsilon} (v_1 L + v_2) + v_3 \right] + \delta$$ \hspace{1cm} (22)

where $\epsilon$ is the broadband surface emissivity; and $\gamma$ and $\delta$ are two parameters given by

$$\gamma = \frac{T^2}{b_\gamma L}$$ \hspace{1cm} (23)

$$\delta = T - \frac{T^2}{b_\gamma}$$ \hspace{1cm} (24)

where $T$ is the at-sensor brightness temperature; $b_\gamma$ is a coefficient equal to 1324K for L8 band 10, or 1199K for band 11. And $v_1$, $v_2$ and $v_3$ are the atmosferic functions, given by

$$v_1 = \frac{1}{\tau_{NB}}; v_2 = -\frac{R_{sky}}{\tau_{NB}} - \frac{R_p}{\tau_{NB}}; v_3 = R_{sky}$$ \hspace{1cm} (25)

where $\tau_{NB}$ is the narrow band transmissivity of air; $R_{sky}$ is the narrow band downward thermal radiation from a clear sky ($W m^{-2} sr^{-1} \mu m^{-1}$); and $R_p$ is path radiance in the 10.4–12.54 $\mu m$ band ($W m^{-2} sr^{-1} \mu m^{-1}$).

3. **method="SW"** (Jimenez-Munoz et al., 2014) (Landsat 8)

$$T_s = T_i + 1.378(T_{10} - T_{11}) + 0.183(T_{10} - T_{11})^2 - 0.268 + (54.30 - 2.238\omega)(1 - \epsilon) + (-129.20 + 16.40\omega)\Delta\epsilon$$ \hspace{1cm} (26)

where $T_s$ is the land surface temperature ($K$); $T_{10}$ and $T_{11}$ are the at-sensor brightness temperatures for bands 10 and 11 of Landsat 8 ($K$); $\epsilon$ is the mean emissivity; $\omega$ is the total atmospheric water vapor content (in g · cm$^{-2}$) and $\Delta\epsilon$ is the emissivity difference.

**Included models for Momentum Roughness Length**

1. **method="short-crops"** (Allen et al., 2007b)

$$Z_{om} = 0.018 \cdot \text{LAI}$$ \hspace{1cm} (27)
2. method="custom" (Allen et al., 2007b)

\[ Z_{om} = \exp((a \times \text{NDVI/albedo}) + b) \]  

(28)

where \(a\) and \(b\) are the regression coefficients derived by adjusting a linear model between \(\log Z_{om} \sim \text{NDVI/albedo}\) for points inside the Landsat scene representing specific vegetation types.

3. method="Perrier" (Santos et al., 2012; Pôças et al., 2014)

\[ Z_{om} = ((1 - \exp(-a \times \text{LAI}/2)) \times \exp(-a \times \text{LAI}/2))^h \]  

(29)

where \(h\) is the crop height in meters, and \(a\) is:

\[ a < -(2 \times (1 - f \text{LAI}))^{-1} \]

when \(f \text{LAI} > 0.5\), or

\[ a < -2 \times f \text{LAI} \]

when \(f \text{LAI} < 0.5\). And \(f \text{LAI}\) is the proportion of LAI lying above \(h/2\).

Guillermo Federico Olmedo  
EEA INTA Mendoza  
San Martín 3853, Luján de Cuyo, Mendoza  
Argentina  
olmedo.guillermo@inta.gob.ar

Samuel Ortega-Farías  
Adaptation of Agriculture to Climate Change (A2C2)  
Centro de Investigación y Transferencia en Riego y Agroclimatología  
Universidad de Talca  
2 Norte 685, Talca  
Chile  
sortega@utalca.cl

Daniel de la Fuente-Sáiz  
Centro de Investigación y Transferencia en Riego y Agroclimatología  
Universidad de Talca  
2 Norte 685, Talca  
Chile  
ddelafuente@utalca.cl

David Fonseca-Luengo  
Centro de Investigación y Transferencia en Riego y Agroclimatología  
Universidad de Talca  
2 Norte 685, Talca  
Chile  
davidtoto.fonseca@gmail.com

Fernando Fuentes-Peñailillo  
Centro de Investigación y Transferencia en Riego y Agroclimatología  
Universidad de Talca  
2 Norte 685, Talca  
Chile  
ffuentesp@utalca.cl