Vegetation Cover and Seasonal Albedo Benchmarking

by Lena Boysen, The Land in the Earth System, MPI-M, Hamburg, 19. Oktober 2011.

Benchmarking, or evaluating models against observations, is a very useful tool to facilitate improvement of the Earth System models. This page describes benchmarking of vegetation dynamics (land cover) as a part of the International Land Model Benchmarking (ILAMB) project (`http://www.ilamb.org/`) as well as benchmarking of seasonal albedo values. The focus is on evaluation of simulated fractions of woody vegetation, bare ground and albedo against observations. Two metrics are used. Spatial correlation between simulated and observed patterns is estimated using Pearson's correlation coefficient (r^2), while the magnitude of a difference between simulations and observations is calculated using the root mean square error ($rmse$). These comparison diagnostic tools are resulting in a scoring system as suggested by Randerson et al. (2009) as well as a visualization of the data sets. The software R is used (`http://www.r-project.org/`) for statistical computing and plotting figures.

Hereafter, a short introduction to the benchmarking scripts is given. By the MODIS data, we mean a MODIS VCF (Vegetation Continuous Fields) product (`http://glcf.umiacs.umd.edu/data/vcf/`), and JSBACH is a land surface model of Max Planck Institute Earth System Model (MPI-ESM). An example is given for the JSBACH vegetation cover simulated in the CMIP5 historical run for the year 2001 at the T63 spatial resolution (ca. $1.9°x1.9°$). The aim is a combination of weighted r^2 with weighted $rmse$. For more detailed instructions, please read the documentation below.

Content of the CORRELATION_ANALYSIS.tar.gz-file

- Detailed documentation of the correlation analysis scripts,
- Scripts for the correlation analysis as listed in figure 1,
- Region masks (GFED2_regions_0.5x0.5.nc.tar.gz),
- Example data of MODIS and JSBACH as well as their results.

Theoretical background

Like explained by Randerson et al. (2009) the spatial correlation between the observation and model data is examined by the calculation of the Pearson's correlation coefficient *r* (eq. 1) using all land grid cells. The scoring is gained by taking the square of *r*.

$$r = \frac{\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \overline{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \overline{Y})^2}} \quad (1)$$

Furthermore, we want to calculate an overall score including tree and desert cover for the tropics and extra-tropics. This is done by taking the mean of the scores (equation 1) for both vegetation types in this two regions. This mean is then weighted by a maximum score (e.g. 5). A similar procedure is done for the root mean square error (*rmse*) (equation 2) which is then combined with the weighted score above by taking the mean of both. The *rmse* between the modeled and the observed data gives the standard deviation of the model prediction error whereby a small value indicates better

model performance. By this an overall score is calculated which allows a comparison of models regarding these two vegetation types.

$$rmse = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(Model_t - Observation_t)^2} \tag{2}$$

Structure of the scripts and usage

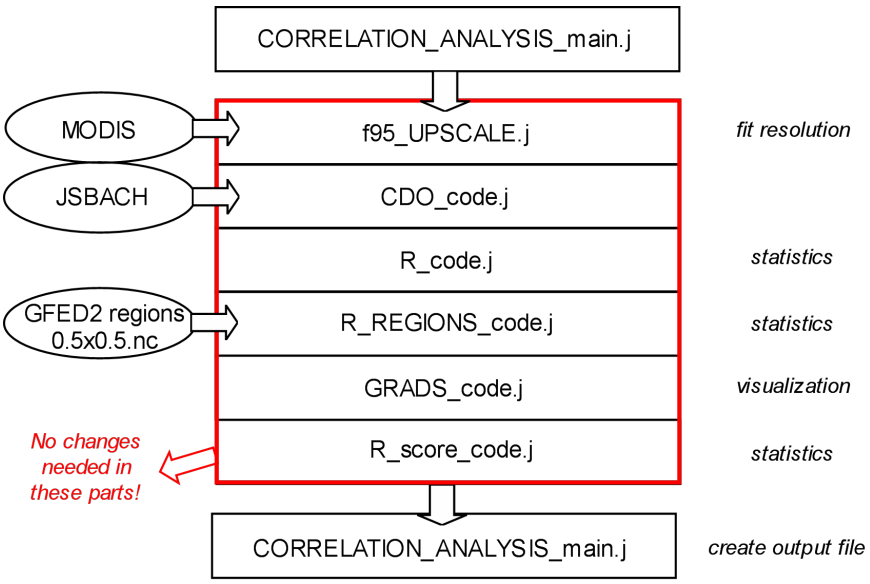

Fig. 1: Structure of the land surface benchmarking scripts.

CORRELATION_ANALYSIS_main.j

This is the navigation file in which all necessary file and variable names are defined as well as their properties are submitted. These declarations are then handed over automatically to the different scripts which modify the data files so *R* can use them and which are responsible for the visualization. This structure is presented in figure 1.
Default calculations are made for the whole globe, the northern and southern hemisphere, the tropics and extra-tropics and the zonal mean. Unless no data files are provided for both, model and observations, the data set for the vegetation type “desert” is gained by subtracting 1-grass-tree which is automatically done by the scripts.
Necessary declarations are:

- The three vegetation types and albedo in this order (renaming possible):
  - grass cover (including crops and pasture),
  - tree cover (including shrubs),
  - desert (provided or generated by the script (1-grass-tree cover)) and
  - albedo (provided or left empty).
- Data files from observation (Ascii or netCDF possible):
  - resolution in latitude and longitude,
  - resolution in degrees

- Data files from model (netCDF):
  - resolution in latitude and longitude,
  - resolution in degrees,
  - year to be computed.
- Weighting factor for the scoring and a
- Folder name for the results (will be created if not existent).

Additionally, the calculations can be extended to the 14 basisregions of the earth (see fig. 2). But since these procedures take some minutes to compute they are optional to chose:

- calculate regions: yes = 1, no = 0,
- path to the GFED2_regions_????_0.5x0.5.nc masks.

Attention:

- If the observation data is also provided as netCDF then please adjust the grid resolution to the model resolution manually and before running the script!
- If the observation data is NOT shifted about 180 degrees to the east, then go to the script f95_UPSCALE.j and set shift=0.
- If the observation data does contain a header, then go to the script f95_UPSCALE.j and set header=1 and hl=line number.

- Version of R: Please check, which version of *R* you use and update it if it is older than **R version 2.10.0** by typing 'module switch R R/2.10.1' for example.
- Necessary R packages: Before starting these scripts, please install two packages in *R* as shown below (Typing "R" in the command window opens the program *R*). Please follow the instructions to create a personal library.

```
R
>> install.packages("gplots")
>> install.packages("hydroGOF")
>> q()
```

f95_UPSCALE.j

The observation data used is provided by e.g. MODIS (Vegetation Continuous Fields data collection, Hansen et al., 2007). Since the data set (provided in Ascii) might have a too fine resolution compared to the model data it must be upscaled so that both data sets are of the same resolution. This script takes the given resolution of the observation data and fits it to the resolution of the model data. The used algorithm averages over a certain amount of grid cells with respect to possible missing values. Furthermore does this Fortran code shift the data set about 180 degrees so that the map begins at 0 and ends at 360 degrees (and not from $180^\circ\ W$ to $180^\circ\ E$). This script is further able to handle a possible header in the data set.

CDO_code.j

This script is responsible for the conversion of the upscaled Ascii-observation-files into netCDF files ($switch = 1$). It therefore uses the grid information of the model data (cdo selgrid). Furthermore it selects the required year from the model data (cdo selyear). The missing values are set uniformly to -99.00 (cdo setmissval) to ensure that *R* will handle them right and will not get confused with too many different declarations. If it is necessary, the data set for the vegetation type "desert" is created here by using the files for grass and tree cover ($switch = 0$). If both data sets of one vegetation

type are provided as netCDF then only the modification of these files takes place ($switch = 2$). Furthermore does this script handle the albedo data, which means that it selects the right seasons (cdo seldate) and the wished variable code (cdo selcode). Seasons are named DJF (winter), MAM (spring), JJA (summer) and SON (autumn). If other codes or dates need to be selected then please change this in CDO_code.j.

R_code.j

Here the statistical computation takes place. This script can also be run alone if the input data sets are already of the same resolution and of the netCDF-format (but the *R*-code can easily be changed to read Ascii as well; read the manual: `http://cran.r-project.org/doc/manuals/R-intro.pdf`). The *R*-code must be run in a mode which saves all data of the session which can then be accessed again later (unfortunately it will list up all steps in the terminal). This is needed to calculate the overall score at the end. Here are the single steps:

1. First of all the package "ncdf" needs to be loaded into the library. The observation and the model data sets are opened and read into seperate .nc-objects which contain all necessary information about the files but not the data itself. The specific variables to be computed are then read into matrices via these .nc-objects where all missing values (-99.00) are replaced by NA (not available, observation data) or NaN (not a number, model data).

```
>> library(ncdf)
>> info.nc = open.ncdf("jsbach.nc")
>> jsbach_matrix = matrix(get.var.ncdf(info.nc, "variable")
>> jsbach_matrix[jsbach_matrix == -99.00] <- NaN (or NA)

The same steps are done for MODIS data!
```

2. Two data vectors must be created without missing values which is done by manual pairwise deletion of NAs and NaNs. Therefore, two new vectors must be defined which have the length of the observation data minus the number of its NA. The data matrices are then transferred to these vectors without the positions where the observation data contains NA. After that, again two new vectors are defined which have the length of the new model data vector minus the number of its NaN. The two data vectors are now transferred to these new vectors without the positions of the model data that contain NaN. Thereby two vectors of the same length and without missing values are created but which still keep the data at the right positions.

```
>> modis_vector <- rep(0.0, times=length(modis_matrix[!is.na[modis_matrix)]))
>> jsbach_vector <- rep(0.0, times=length(modis_matrix[!is.na[modis_matrix)]))

#Loop over all grid points [i,j] in matrices:
k=1
for(i in 1:length(latitudes)){
for(j in 1:length(longitudes)){

  if(!is.na(modis_matrix[i,j])){
modis_vector[k] <- modis_matrix[i,j]
jsbach_vector[k] <- jsbach_matrix[i,j]}
k=k+1}
}}

>> jsbach_vector <- rep(0.0, times=length(jsbach_matrix[!is.na[jsbach_matrix)]))
>> modis_vector <- rep(0.0, times=length(jsbach_matrix[!is.na[jsbach_matrix)]))
```

```
# Loop over all positions [p] in vectors:
m=1
for(p in 1:(k-1)){
  if(!is.nan(jsbach_vector[p])){
jsbach_vector2[m] <- jsbach_vector[p]
modis_vector2[m] <- modis_vector[p]
m=m+1}
}
```

3. Here the calculation of the correlation coefficient takes place. The command below calculates the Pearson correlation coefficient (eq. 1) between two data vectors, whereby use="all.obs" gives an error if there are still missing values. The score (square of *r*) is calculated afterwards.

```
>> correlation_coefficient <- cor(modis_vector2,jsbach_vector2,use="all.obs")
>> score_cor <- correlation_coefficient*correlation_coefficient
```

4. The root mean square error (eq. 2) is calculated by using the package "hydroGOF" which needs to be loaded into the library as well.

```
>> library(hydroGOF)
>> RMSE <- rmse(jsbach_vector2,modis_vector2,use="all.obs")
```

5. This calculation is done for the whole globe, the northern hemisphere, the southern hemisphere, the tropics and the extra-tropics for which only the necessary latitudes are read into the matrix.
6. The score (square of *r*) and the correlation coefficients are then listed in a table by using a data.frame and a textplot (package "gplots" neccessary).

```
>> library(gplots)
>> cor_data <- data.frame(c(header),c(correlation-coefficients),c(scores_cor))
>> textplot(cor_data,halign="center",valign="center"...)
>> title("Pearsons's correlation coefficients")
```

7. Scatterplots demonstrate the correlation between the two data sets with the simple command "plot". A graph of the linear regression serves for orientation. Therefore, the linear regression is calculated with "lm" and a graph is plotted with "abline" according to this result. A "best fit" line is drawn afterwards where "lty" changes the linetype.

```
>> plot(modis_vector2,jsbach_vector2,main="Title",xlab="MODIS",ylab="JSBACH"...)
>> myline.fit <- lm(jsbach_vector2 ~ modis_vector2)
>> summary(myline.fit)
>> abline(myline.fit,col="red",lty=4)
>> abline(a=0,b=1,lty=4)
```

8. Furthermore is the zonal mean calculated and displayed. First, vectors have to be created which are of the length of all latitudinal points. "na.rm=TRUE" makes sure that missing values are not taken into account when computing the mean. The x-axis limits and ticks are handed over by the main script.

```
>> modis_zonal <- rep(0.0,times=length(latitudes) (or modis_zonal = NULL is also possible)
>> jsbach_zonal <- rep(0.0,times=length(latitudes)

# Loop over all latitudes:
k=1
for(i in 1:length(lati)){
```

```
    modis_zonal[k] <- mean(modis_matrix[i,],na.rm=TRUE)
    jsbach_zonal[k] <- mean(jsbach_matrix[i,],na.rm=TRUE)
    k=k+1
}

>> plot(c(latitudes),modis_zonal,col="red"...)
>> lines(c(latitudes),jsbach_zonal,col="blue"'...)
```

R_REGIONS_code.j

If the calculation of the 14 basisregions as shown in figure 2 is switched on, this script then computes the correlation coefficients and the scores by multiplying the matrices in *R* with region masks provided by the GFED2_regions_0.5x0.5.nc (Global Fire Emissions Database, 2011) files. This loop takes some minutes since the masks have to be upscaled as well (cdo remapnn) using the grid information of the model data. The script will check if the maps have already been upscaled once so that this procedure is only done for the first time the script is run. The results are listed in a table in an Ascii- file which is converted to .pdf later on. No visualization is provided.

BONA	Boreal North America	NHAF	Northern Hemisphere Africa
TENA	Temperate North America	SHAF	Southern Hemisphere Africa
CEAM	Central America	BOAS	Boreal Asia
NHSA	Northern Hemisphere South America	CEAS	Central Asia
SHSA	Southern Hemisphere South America	SEAS	Southeast Asia
EURO	Europe	EQAS	Equatorial Asia
MIDE	Middle East	AUST	Australia and New Zealand

Fig. 2: Location of the 14 basisregions. Source: `http://globalfiredata.org/Tables/index.html`

GRADS_code.j

To get a better impression of the data sets the land cover is plotted here with GrADS (`http://www.iges.org/grads/`). The titles of the figures can be changed in the script itself. The upper part of the script is responsible for the creation of a colorbar which is switch aroung if the vegetation type is „desert“. Additionally is the difference between the model and the observation data plottet. Output can also be changed into .png (default is .pdf) by uncomment the command 'printim name.png' in the code. Colorbars and their levels can be adjusted here as well.

R_score_code.j

This script accesses the stored results *R*. This is necessary to average the values of the correlation scores and the root mean square errors for the vegetation types tree and desert in the regions of the tropics and extratropics. These means are then weighted by a maximum factor (e.g. 5) and combined to one single overall score for the model (equation (3)). Albedo is not yet included in this calculation.

$$score = \frac{1}{2}\Big(5 \cdot mean(r^2) + 5 \cdot [1 - mean(rmse)]\Big) \tag{3}$$

```
>> score_corr <- maxscore*mean(c(score_cor_tropics,score_cor_extratropics...))
>> score_rmse <- maxscore*mean(c(RMSE_tropics,RMSE_extratropics...))

>> TOTAL_SCORE <- mean(c(score_corr,(1-score_rmse)))
```

More information on this procedure is given by Abramowitz et al (2008). These values are again printed into a pdf-file by using data.frame and textplot.

CORRELATION_ANALYSIS_main.j

Finally, the resulting files from *R* and GrADS can be combined to one .pdf file (but *convert* results in a loss of quality!). If the naming is not appropriate please change it in the last command in the lines where you can find: *** here the final OUTPUT is named ***.
The results can be automatically displayed with xpdf. All unused files are deleted within each script and at the very end of the main script which is important if for example the upscaled netCDFs shall be saved. All results are moved into a file directory which is specified at the beginning of the main script (result_dir).

References

Abramowitz G., Leuning R., Clark M., Pitman A. (2008): Evaluating the Performance of Land Surface Models. American Meteorological Society, Journal of Climate, Volume 21, DOI: 10.1175/2008JCLI2378.1.

Global Fire Emissions Database, URL: `http://globalfiredata.org/Data/index.html`, accessed June 9, 2011.

Hansen M., R. DeFries, J. R. Townshend, M. Carroll, C. Dimiceli, and R. Sohlberg (2007): 2001 Percent Tree Cover, Collection 4, Vegetation Continuous Fields MOD44B, `http://www.landcover.org/data/vcf/`, Univ. of Md., College Park.

Randerson J. T., Hoffman F. M., Thornton P.E., Mahowald N. M., Lindsay K., Lee Y.-H., Nevison C. D., Doney S. C., Bonan G., Stöckli R., Covey C., Running S. W., Fung I. (2009): Systematic Assessment of Terrestrial Biogeochemistry in Coupled ClimateCarbon Models. Global Change Biology, Volume 15, Issue 9, DOI: 10.1111/j.13652486.2009.01912.x.