Supplement of Biogeosciences Discuss., 12, 51–92, 2015
http://www.biogeosciences-discuss.net/12/51/2015/
doi:10.5194/bgd-12-51-2015-supplement

Biogeosciences
Discussions
Open Access

Supplement of

Assessing vegetation structure and ANPP dynamics in a grassland-shrubland Chihuahuan ecotone using NDVI-rainfall relationships

M. Moreno-de las Heras et al.

Correspondence to: M. Moreno-de las Heras (mariano.moreno-de-las-heras@durham.ac.uk)

In this document we provide the Maple 9.5 (Maplesoft, Waterloo, Canada) codes used in the paper (Code 1) to simulate dryland biomass dynamics for an herbaceous and a shrub species, and (Code 2) to decompose single time series of NDVI into partial components for herbaceous and shrub vegetation applying the reference vegetation-type characteristic antecedent rainfall series for herbs and shrubs ($ARain_{hv}$ and $ARain_{s}$, respectively). We also provide a supplementary figure (Supplementary Fig. 1) that presents detailed NDVI-antecedent rainfall correlograms obtained for each growing cycle of vegetation growth (April-March) in the reference Black grama and Creosotebush SEV LTER Core Sites.

Contents:

Code 1: Dynamic Vegetation Model

Input files (location: C:\DataFolder\)**:**

1. Daily rainfall: Rain.txt

Data is stored in columns 1 and 2 for dates and rainfall, respectively.

Output files (location: C:\DataFolder\)**:**

1. Temporal series of herbaceous and shrub biomass: Biomass.txt

Data is stored in columns 1, 2 and 3 for dates, herbaceous and shrub biomass, respectively.

2. Temporal series of herbaceous and shrub biomass graph: Biomass.png (green, herbaceous biomass; red, shrub biomass; blue, daily rainfall).

Procedure:

1. We load the Maple packages required for the subsequent calculations.

```
> with(linalg): with(plots): with(LinearAlgebra): with(Statistics): with(plottools):
```

2. We load the daily rainfall data file.

```
> droot := "C:\\DataFolder\\":
 drain := ImportMatrix(cat(droot, "Rain.txt"), source = delimited, delimiter = "  ", datatype
= anything):
dates := ImportMatrix(cat(droot, "Rain.txt"), source = delimited, delimiter = "",
datatype=string):
```

3. We define a rainfall function (rainFunct) made by rainfall event pulses.

```
> rainn := convert(Column(drain, 2), list):
 revent := [NULL]; raint := 0:
 for i to nops(rainn) do
 prec := convert(rainn[i], float):
```

```
if prec > 0 then
revent := [op(revent), [i, prec]]:
raint := raint+prec:
fi:
od:
rainFunct := t→sum(revent[jjk][2]*(-Heaviside(t-revent[jjk][1])+Heaviside(t-revent[jjk][1]+1)), jjk = 1 .. nops(revent)):
ndata := nops(rainn);
```

4. We define the model equations.

```
> dB := gmax*(W-W0)*B/(W+kw)-m*B;
dW := P*(B+ki*i0)/(B+ki)-c*gmax*(W-W0)*B/(W+kw)-rw*W;
dsys := subs(W = W(t), B = B(t), [dB, dW]):
ecdif := [diff(B(t), t) = dsys[1], diff(W(t), t) = dsys[2]]:
```

5. We define a time-evolution function (evolution) that calculates and stores biomass values for each day, integrating the model equations with the model parameter values.

```
> evolution := proc (param)
local stot, Biomasst, i:
stot := dsolve({op(subs(P = rainFunct(t), param, ecdif)), B(0) = 50, W(0) = .2}, numeric, maxfun = 0):
Biomasst := NULL:
for i to ndata do
Biomasst := op([Biomasst]), subs(stot(i), B(t)):
od:
RETURN(Biomasst)
```

```
end proc:
```

6. We define the parameter values and call the time-evolution function.

```
> herbParam := W0 = 0.05, kw = 0.45, ki = 180, i0 = 0.2, c = 0.1, rw = 0.1, gmax = 0.32, m = 0.05:
shrubParam := W0 = 0.05, kw = 0.45, ki = 180, i0 = 0.2, c = 0.1, rw = 0.1, gmax = 0.12, m = 0.03:
herbBiomass := evolution({herbParam}):
shrubBiomass := evolution({shrubParam}):
```

7. We plot the time series of herbaceous and shrub biomass along with precipitation.

```
> topl := 700:
figherb := pointplot([seq([i, herbBiomass[i]], i = 1 .. nops([herbBiomass]))], connect = true, color = green):
figshrub := pointplot([seq([i, shrubBiomass[i]], i = 1 .. nops([shrubBiomass]))], connect = true, color = red):
figYears := [NULL]:
for iy to 16 do
figYears := [op(figYears), pointplot([[365*iy, 0], [365*iy, topl]], color = grey, connect = true, linestyle = 3)]
od:
figPrecipt := NULL:
for i to ndata do if drain[i][2] > 0 then
figPrecipt := op([figPrecipt]), pointplot([[i, topl], [i, topl-4*drain[i][2]]], connect = true, color = navy, thickness = 3):
fi:
od:
```

```
figures:= display(figherb, figshrub, figYears, figPrecipt):
display(figures);
```

8. We export the output files.

```
fout := cat(droot, "Biomass.txt"):
for i to ndata do
FileTools[Text][WriteLine](fout, cat(dates[i][1], " ", convert(herbBiomass[i], string), " ", convert(shrubBiomass[i], string))):
od:
FileTools[Text][Close](fout):
plotsetup(png, plotoutput = cat(droot, "Biomass.png")):
display(figures);
plotsetup(default):
```

Code 2: NDVI Decomposition Procedure

Input files (location: C:\DataFolder\)**:**

1. NDVI experimental data: case.txt

Data is stored in column 1.

2. Characteristic antecedent rainfall series for herbaceous and shrub vegetation ($ARain_{hv}$ and $ARain_s$, respectively): totalAR.txt

Data is stored in columns 1 and 2 for herbaceous and shrub vegetation, respectively.

3. Time in days from the initial date: totalT.txt

Data is stored in column 1.

Output files (location: C:\DataFolder\)**:**

1. Temporal series of herbaceous and shrub NDVI components: HScomponents.txt

Data is stored in columns 1 and 2 for herbaceous and shrub biomass, respectively.

2. Graph with the temporal series of herbaceous and shrub NDVI, along with the original total NDVI signal: HScomponents.png (black, original signal; green, herbaceous component; red, shrub component).

Procedure:

1. We load the Maple packages required for the subsequent calculations.

```
> with(ExcelTools): with(plots): with(plottools): with(LinearAlgebra): with(Statistics):
```

2. We define the NDVI bare soil component (0.12) and define a function, pair, to handle data lists.

```
nsoil := 0.12;

pair := proc (x, y)
[x, y]
end proc
```

2. We load the data files and store data as lists. The following data lists are defined:

dataAR1 = antecedent rainfall series for herbaceous vegetation (57-day period, $ARain_{hv}$ series *).*

dataAR2 = antecedent rainfall series for shrubs (145-day period, $ARain_s$ series*).*

dataT = time (measured in days from the beginning of the series).

dataNDVI = original NDVI time series.

dataNDVI0 = NDVI data list without the soil base line.

```
> droot := "C:\\ DataFolder \\":
 dNDVI := ImportMatrix(cat(droot, "case.txt"), source = delimited, delimiter = "          ",
datatype = anything):
 totalAR := ImportMatrix(cat(droot, "TotalAR.txt"), source = delimited, delimiter = "   ",
datatype = anything):
 totalT := ImportMatrix(cat(droot, "totalT.txt"), source = delimited, delimiter = " "):
 Ndata := op(rtable_dims(dNDVI)[1])[2]:
 dataAR1 := [NULL]: dataAR2 := [NULL]: dataAR1N := [NULL]: dataAR2N := [NULL]:
 dataT := [NULL]: dataNDVI := [NULL]: dataNDVI0 := [NULL]:
 for i to Ndata do
dataAR1 := [op(dataAR1), evalf(totalAR[i][1])]; dataAR2 := [op(dataAR2),
evalf(totalAR[i][2])]; dataT := [op(dataT), evalf(totalT[i][1])]; dataNDVI :=
[op(dataNDVI), evalf(dNDVI[i][1])]; dataNDVI0 := [op(dataNDVI0), evalf(dNDVI[i][1]-
nsoil)]
od:
```

4. We define a first-order least-squares optimization function (linearfit) that fits the partial contribution of the herbaceous and shrub components to the time series of NDVI (filtered for the base-line bare soil contribution, dataNDVI0) as a function of the vegetation-type specific *antecedent rainfall series that maximize the NDVI-precipitation relationships for herbaceous vegetation (dataAR1,* $ARain_{hv}$ series*) and for shrubs (dataAR2,* $ARain_s$ series*).*

```
>linearfit := proc (TAR1, TAR2, Tiemp, NDVIst)
local AInput, DOutput, fitlinear, dparam, i, sumres;
global Total;
AInput := zip(pair, TAR1, TAR2); DOutput := NDVIst;
fitlinear := LinearFit([ar1, ar2], AInput, DOutput, [ar1, ar2], output = solutionmodule);
dparam := fitlinear:-Results("leastsquaresfunction"); sumres := fitlinear:-
Results("residualsumofsquares");
Total := [NULL]; for i to Ndata do Total := [op(Total), subs(ar1 = AInput[i][1], ar2 =
AInput[i][2], dparam+nsoil)] od:
RETURN(dparam, sumres):
end proc:
```

5. We define a function that reassigns the predicted weights of the fitted vegetation components (i.e. the percentage contribution of each vegetation type over the predicted totals for any t_i*) to match the original shape of the NDVI time series, obtaining the final NDVI components for herbaceous vegetation and shrubs.*

```
> linDecomp := proc (TAR1, TAR2, NDVIst, fit)
local Ntotal, j, i, pre1, pre2, ratio;
global Nherb, Nshrub;
Nherb := [NULL]; Nshrub := [NULL]; Ntotal := [NULL];
for i to Ndata do
pre1 := subs(ar1 = TAR1[i], ar2 = 0, fit); pre2 := subs(ar1 = 0, ar2 = TAR2[i], fit);
if 0 <= pre1 and 0 <= pre2 then ratio := NDVIst[i]/subs(ar1 = TAR1[i], ar2 = TAR2[i], fit);
Ngrass := [op(Nherb), pre1*ratio]; Nshrub := [op(Nshrub), pre2*ratio] elif pre1 < 0 and 0
<= pre2 then Nherb := [op(Nherb), 0]; Nshrub := [op(Nshrub), NDVIst[i]] elif pre2 < 0
and 0 <= pre1 then Nherb := [op(Nherb), NDVIst[i]]; Nshrub := [op([Nshrub]), 0] else
print(errors); ratio := 1; Nherb := [op(Nherb), 0]; Nshrub := [op(Nshrub), 0] fi;
Ntotal := [op(Ntotal), Nherb[nops(Nherb)]+Nshrub[nops(Nshrub)]+nsoil] od;
 RETURN(Nherb, Nshrub, Ntotal):
end proc:
```

6. We call the fitting and reassigning functions.

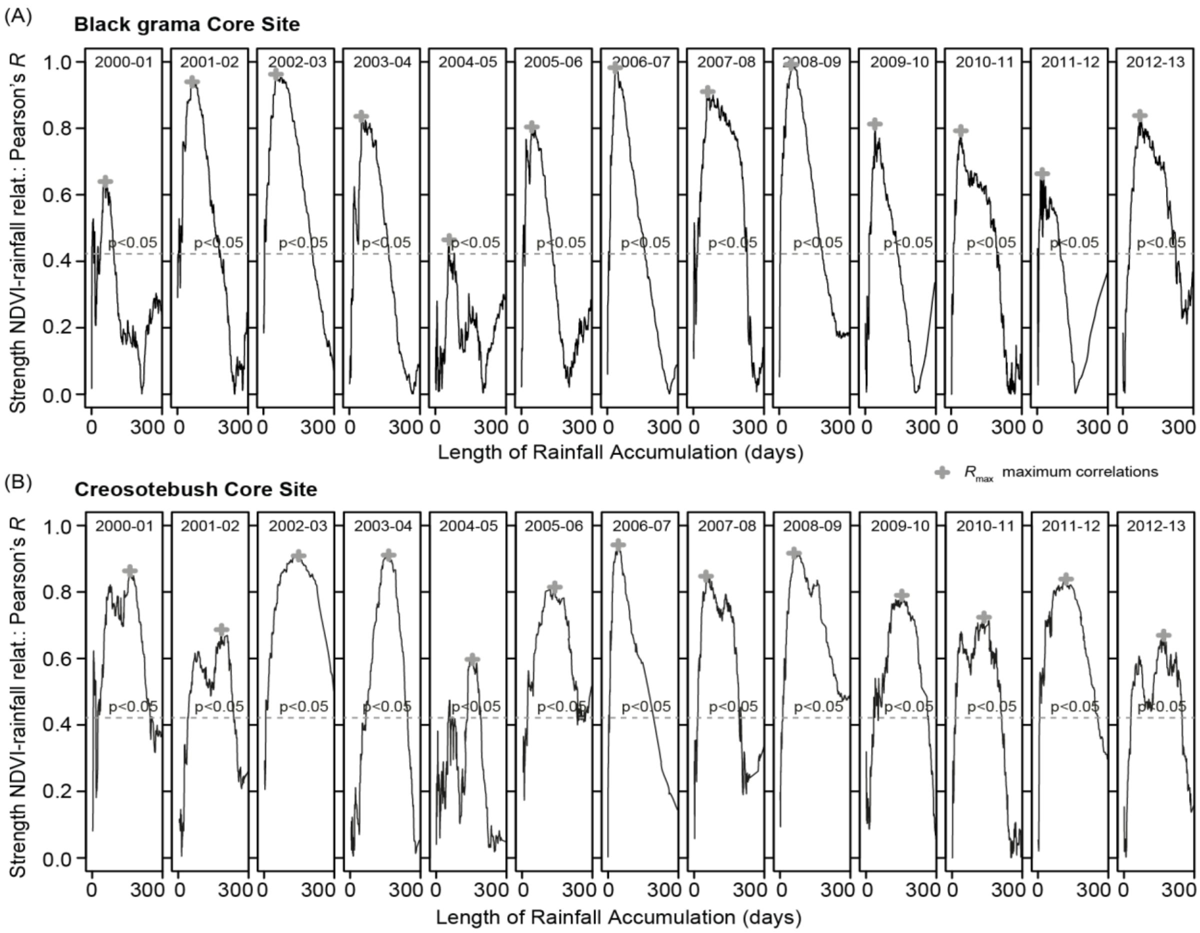
```
lfit1 := linearfit(dataAR1, dataAR2, dataT, dataNDVI0);
HerbShrubLineal := linDecomp(dataAR1, dataAR2, dataNDVI0, lfit1[1]):
```

7. We plot the time series of the NDVI signal (figOr), and the final NDVI components for herbaceous vegetation (figHerb) and shrubs (figShrub).

```
figOr := PLOT(CURVES(convert(sort(zip(pair, dataT, dataNDVI)), list))):
figHerb := PLOT(CURVES(sort(sort(zip(pair, dataT, Nherb)))), COLOR(RGB, 0, 1, 0)):
figShrub := PLOT(CURVES(sort(sort(zip(pair, dataT, Nshrub)))), COLOR(RGB, 1, 0, 0)):
display(figOr, figHerb, figShrub);
```

8. We export the output files.

```
fout := cat(droot, "HScomponents.txt"):
for i to Ndata do
FileTools[Text][WriteLine](fout, cat(convert(Nherb[i], string), " ", convert(Nshrub[i],
string))):
od:
FileTools[Text][Close](fout):
plotsetup(png, plotoutput = cat(droot, "HScomponents.png")):
display(figOr, figHerb, figShrub):
plotsetup(default):
```



Supplementary Fig. 1. Per annual growing cycle (April-March) NDVI-antecedent rainfall correlograms for the **(A)** Black grama and **(B)** Creosotebush SEV LTER Core Sites.

Notes:

Correlations between NDVI and antecedent precipitation are maximized using a rainfall accumulation length of about 57 days for all annual cycles of vegetation growth in the Black grama Core Site (Supplementary Fig. 1A).

For the Creosotebush Core Site two different foci that maximize the correlation between NDVI and antecedent rainfall can be detected: (i) one using a low rainfall accumulation length (about 57 days) and (ii) another using a long rainfall accumulation length (about 145 days). The 145 days antecedent rainfall series generally shows a stronger correlation with the NDVI than the 57 days antecedent rainfall series (cycles 2000-01, 2001-02, 2002-03, 2003-04, 2004-05, 2009-10, 2010-11, 2011-12, 2012-13). However, for three consecutive annual cycles with strong summer precipitation (2006-07, 2007-08, and 2008-09, summer precipitation for the period is 40% above the long-term mean) correlation of NDVI to the the 57 days antecedent rainfall series is stronger that to the 145 days antecedent rainfall series (Supplementary Fig. 1B).