Procesamiento de series de tiempo en GRASS GIS

Aplicaciones en Ecología y Ambiente

Dra. Verónica Andreo
CONICET - INMeT

Río Cuarto, 2018

Landscape, hydrology and terrain analysis

Overview

- Landscape structure analysis and forest fragmentation
- Hydrology analysis: Estimating inundation extent using HAND methodology
- Terrain analysis with geomorphons

Download the file with code to follow the exercise

Landscape structure analysis and forest fragmentation

Richness and Diversity

```
#!/bin/bash
############################################################################
# Exercises for raster data processing in GRASS GIS
# GRASS GIS postgraduate course in Rio Cuarto
# Author: Veronica Andreo
# October, 2018
############################################################################



#
# Landscape analysis and forest fragmentation
#



# install addons
g.extension r.diversity
g.extension r.forestfrag
```

Richness and Diversity

```
############################################################################
#
# Landscape analysis and forest fragmentation
#


# install addons
g.extension r.diversity
g.extension r.forestfrag

# compute richness (number of unique classes)
g.region raster=landclass96
r.neighbors input=landclass96 output=richness method=diversity size=15

# compute diversity indices with various window sizes
r.diversity input=landclass96 prefix=index alpha=0.8 size=9-21 method=s
```

Install addons

Richness and Diversity

```
# Landscape analysis and forest fragmentation
#



# install addons
g.extension r.diversity
g.extension r.forestfrag

# compute richness (number of unique classes)
g.region raster=landclass96
r.neighbors input=landclass96 output=richness method=diversity size=15

# compute diversity indices with various window sizes
r.diversity input=landclass96 prefix=index alpha=0.8 size=9-21 method=s

# change color tables to make them comparable
r.colors map=index_shannon_size_21,index_shannon_size_15,index_shannon_
r.colors map=index_renyi_size_21_alpha_0.8,index_renyi_size_15_alpha_0.
# we use grey.1.0 color ramp because simpson is from 0 to 1
```

Estimate richness - The window size is in cells

Richness and Diversity

```
# install addons
g.extension r.diversity
g.extension r.forestfrag

# compute richness (number of unique classes)
g.region raster=landclass96
r.neighbors input=landclass96 output=richness method=diversity size=15

# compute diversity indices with various window sizes
r.diversity input=landclass96 prefix=index alpha=0.8 size=9-21 method=s

# change color tables to make them comparable
r.colors map=index_shannon_size_21,index_shannon_size_15,index_shannon_
r.colors map=index_renyi_size_21_alpha_0.8,index_renyi_size_15_alpha_0.
# we use grey1.0 color ramp because simpson is from 0 to 1
r.colors map=index_simpson_size_21,index_simpson_size_15,index_simpson_
```

Compute Simpson, Shannon, and Renyi diversity indices

Richness and Diversity

```
# compute richness (number of unique classes)
g.region raster=landclass96
r.neighbors input=landclass96 output=richness method=diversity size=15

# compute diversity indices with various window sizes
r.diversity input=landclass96 prefix=index alpha=0.8 size=9-21 method=s

# change color tables to make them comparable
r.colors map=index_shannon_size_21,index_shannon_size_15,index_shannon_
r.colors map=index_renyi_size_21_alpha_0.8,index_renyi_size_15_alpha_0.
# we use grey1.0 color ramp because simpson is from 0 to 1
r.colors map=index_simpson_size_21,index_simpson_size_15,index_simpson_


#
# Forest fragmentation
#
```

Make colors comparable

***Task**: Add all maps to Map Display using Add multiple raster or vector layers in Layer manager toolbar (top).*

- How do different indices compare to each other?
- How does changing window size affect the diversity measure?

Forest fragmentation

We'll use the addon r.forestfrag that computes the forest fragmentation following the methodology proposed by Riitters et al. (2000).

Forest fragmentation

```
#!/bin/bash
############################################################################
# Exercises for raster data processing in GRASS GIS
# GRASS GIS postgraduate course in Rio Cuarto
# Author: Veronica Andreo
# October, 2018
############################################################################



#
# Landscape analysis and forest fragmentation
#



# install addons
g.extension r.diversity
g.extension r.forestfrag
```

Forest fragmentation

```
r.colors map=index_simpson_size_21,index_simpson_size_15,index_simpson_


#
# Forest fragmentation
#



# first set region
g.region raster=landclass96

# list classes:
r.category map=landclass96

# select forests only
r.mapcalc "forest = if(landclass96 == 5, 1, 0)"

# compute the forest fragmentation index with window size 15
```

Set the computational region

Forest fragmentation

```
#
# Forest fragmentation
#



# first set region
g.region raster=landclass96

# list classes:
r.category map=landclass96

# select forests only
r.mapcalc "forest = if(landclass96 == 5, 1, 0)"

# compute the forest fragmentation index with window size 15
r.forestfrag input=forest output=fragmentation window=15

# distribution of the fragmentation categories
```

List the classes in LULC map

Forest fragmentation

```
# first set region
g.region raster=landclass96

# list classes:
r.category map=landclass96

# select forests only
r.mapcalc "forest = if(landclass96 == 5, 1, 0)"

# compute the forest fragmentation index with window size 15
r.forestfrag input=forest output=fragmentation window=15

# distribution of the fragmentation categories
r.report map=fragmentation units=k,p
```

Create new map with forest class only

Forest fragmentation

```
g.region raster=landclass96

# list classes:
r.category map=landclass96

# select forests only
r.mapcalc "forest = if(landclass96 == 5, 1, 0)"

# compute the forest fragmentation index with window size 15
r.forestfrag input=forest output=fragmentation window=15

# distribution of the fragmentation categories
r.report map=fragmentation units=k,p



#
# Distance from forest edge
#
```

Compute the forest fragmentation index

Forest fragmentation

```
r.category map=landclass96

# select forests only
r.mapcalc "forest = if(landclass96 == 5, 1, 0)"

# compute the forest fragmentation index with window size 15
r.forestfrag input=forest output=fragmentation window=15

# distribution of the fragmentation categories
r.report map=fragmentation units=k,p


#
# Distance from forest edge
#


# select forests only
```

Report the distribution of fragmentation categories

***Task**: Explore the effect of different window sizes over fragmentation categories.*

Further details: blog about r.forestfrag

Distance from forest edge

```
#!/bin/bash
############################################################################
# Exercises for raster data processing in GRASS GIS
# GRASS GIS postgraduate course in Rio Cuarto
# Author: Veronica Andreo
# October, 2018
############################################################################



#
# Landscape analysis and forest fragmentation
#



# install addons
g.extension r.diversity
g.extension r.forestfrag
```

Distance from forest edge

```
r.report map=fragmentation units=k,p


#
# Distance from forest edge
#



# select forests only
r.mapcalc "forest = if(landclass96 == 5, 1, null())" --o

# get distance to the edge of the forest
# -n is to obtain distance to the edge from within the forest itself
r.grow.distance -n input=forest distance=distance


#
# Patch analysis
```

Create new map with forest class only - note null function

Distance from forest edge

```
#
# Distance from forest edge
#



# select forests only
r.mapcalc "forest = if(landclass96 == 5, 1, null())" --o

# get distance to the edge of the forest
# -n is to obtain distance to the edge from within the forest itself
r.grow.distance -n input=forest distance=distance


#
# Patch analysis
#
```

Get distance from center to forest edge - note -n flag

***Task**: Display raster map obtained and get univariate statistics*

Landscape patch analysis
Set the config file in the g.gui.rlisetup config window

- Hit "Create" and name the config file *forest_whole*
- Select the raster map forest
- Define the sampling region --> whole map layer
- Define sample area --> whole map layer

- Hit Create and name the config file *forest_mov_win*
- Select the raster map forest
- Define the sampling region --> whole map layer
- Define sample area --> moving window
- Select shape of mov window --> rectangle --> width=10, height=10

Landscape patch analysis

```
#!/bin/bash
############################################################################
# Exercises for raster data processing in GRASS GIS
# GRASS GIS postgraduate course in Rio Cuarto
# Author: Veronica Andreo
# October, 2018
############################################################################



#
# Landscape analysis and forest fragmentation
#



# install addons
g.extension r.diversity
g.extension r.forestfrag
```

Landscape patch analysis

```
# 1. Create
# 2. Name the config file `forest_mov_win`
# 3. Select the raster map forest
# 4. Define the sampling region --> whole map layer
# 5. Define sample area ==> moving window
# 6. Select shape of mov window --> rectangle --> width=10, height=10


# Compute the landscape metrics using both config files


# edge density
r.li.edgedensity input=forest config=forest_whole output=forest_edge_fu
# shape index
r.li.shape input=forest config=forest_whole output=forest_shape_full
# patch number
r.li.patchnum input=forest config=forest_whole output=forest_patchnum_f
# mean patch size
r.li.mps input=forest config=forest_whole output=forest_mps_full

# edge density
```

Compute edge density for the whole area

Landscape patch analysis

```
# 3. Select the raster map forest
# 4. Define the sampling region --> whole map layer
# 5. Define sample area --> moving window
# 6. Select shape of mov window --> rectangle --> width=10, height=10

# Compute the landscape metrics using both config files

# edge density
r.li.edgedensity input=forest config=forest_whole output=forest_edge_fu
# shape index
r.li.shape input=forest config=forest_whole output=forest_shape_full
# patch number
r.li.patchnum input=forest config=forest_whole output=forest_patchnum_f
# mean patch size
r.li.mps input=forest config=forest_whole output=forest_mps_full

# edge density
r.li.edgedensity input=forest config=forest_mov_win output=forest_edge_
# shape index
```

Compute shape index for the whole area

Landscape patch analysis

```
# 5. Define sample area --> moving window
# 6. Select shape of mov window --> rectangle --> width=10,,height=10

# Compute the landscape metrics using both config files

# edge density
r.li.edgedensity input=forest config=forest_whole output=forest_edge_fu
# shape index
r.li.shape input=forest config=forest_whole output=forest_shape_full
# patch number
r.li.patchnum input=forest config=forest_whole output=forest_patchnum_f
# mean patch size
r.li.mps input=forest config=forest_whole output=forest_mps_full

# edge density
r.li.edgedensity input=forest config=forest_mov_win output=forest_edge_
# shape index
r.li.shape input=forest config=forest_mov_win output=forest_shape_mw
# patch number
```

Compute patch number for the whole area

Landscape patch analysis

```
# Compute the landscape metrics using both config files

# edge density
r.li.edgedensity input=forest config=forest_whole output=forest_edge_fu
# shape index
r.li.shape input=forest config=forest_whole output=forest_shape_full
# patch number
r.li.patchnum input=forest config=forest_whole output=forest_patchnum_f
# mean patch size
r.li.mps input=forest config=forest_whole output=forest_mps_full

# edge density
r.li.edgedensity input=forest config=forest_mov_win output=forest_edge_
# shape index
r.li.shape input=forest config=forest_mov_win output=forest_shape_mw
# patch number
r.li.patchnum input=forest config=forest_mov_win output=forest_patchnum
# mean patch size
```

Compute mean patch size for the whole area

***Task**: Now, do the same for the moving window case and compare outputs and results.*

For an overview of r.li.* modules, see: r.li manual

Notes:

If the "moving window" method was selected in g.gui.rlisetup, the output will be a raster map, otherwise an ASCII file will be generated in the folder `C:\Users\userxy\AppData\Roaming\GRASS7\r.li\output\` (MS-Windows) or `$HOME/.grass7/r.li/output/` (GNU/Linux).

Hydrology: Estimating inundation extent using HAND methodology

- r.watershed: for computing flow accumulation, drainage direction, the location of streams and watershed basins
- r.lake: fills a lake to a target water level from a given start point or seed raster
- r.lake.series: addon which runs r.lake for different water levels
- r.stream.distance: for computing the distance to streams or outlet, the relative elevation above streams

Inundation extent using HAND methodology

```
#!/bin/bash
############################################################################
# Exercises for raster data processing in GRASS GIS
# GRASS GIS postgraduate course in Rio Cuarto
# Author: Veronica Andreo
# October, 2018
############################################################################



#
# Landscape analysis and forest fragmentation
#



# install addons
g.extension r.diversity
g.extension r.forestfrag
```

Inundation extent using HAND methodology

```
r.li.mps input=forest config=forest_mov_win output=forest_mps_mw



#
# Hydrology: Estimating inundation extent using HAND methodology
#



# install r.stream.distance and r.lake.series
g.extension r.stream.distance
g.extension r.lake.series

# set the computational region
g.region -p raster=elevation

# compute the flow accumulation, drainage and streams (with threshold v
r.watershed elevation=elevation accumulation=flowacc \
 drainage=drainage stream=streams threshold=100000
```

Install required addons

Inundation extent using HAND methodology

```
#
# Hydrology: Estimating inundation extent using HAND methodology
#



# install r.stream.distance and r.lake.series
g.extension r.stream.distance
g.extension r.lake.series

# set the computational region
g.region -p raster=elevation

# compute the flow accumulation, drainage and streams (with threshold v
r.watershed elevation=elevation accumulation=flowacc \
 drainage=drainage stream=streams threshold=100000

# convert the streams to vector for better visualization
r.to.vect input=streams output=streams type=line
```

Set computational region

Inundation extent using HAND methodology

```
# install r.stream.distance and r.lake.series
g.extension r.stream.distance
g.extension r.lake.series

# set the computational region
g.region -p raster=elevation

# compute the flow accumulation, drainage and streams (with threshold v
r.watershed elevation=elevation accumulation=flowacc \
 drainage=drainage stream=streams threshold=100000

# convert the streams to vector for better visualization
r.to.vect input=streams output=streams type=line

# compute new raster with elevation difference between
# the cell and the the cell on the stream where the cell drains
r.stream.distance stream_rast=streams direction=drainage \
```

Compute the flow accumulation, drainage and streams

Inundation extent using HAND methodology

```
g.extension r.lake.series

# set the computational region
g.region -p raster=elevation

# compute the flow accumulation, drainage and streams (with threshold v
r.watershed elevation=elevation accumulation=flowacc \
 drainage=drainage stream=streams threshold=100000

# convert the streams to vector for better visualization
r.to.vect input=streams output=streams type=line

# compute new raster with elevation difference between
# the cell and the the cell on the stream where the cell drains
r.stream.distance stream_rast=streams direction=drainage \
 elevation=elevation method=downstream difference=above_stream

# compute a lake from specified coordinates and water level
r.lake elevation=elevation water_level=90 lake=lake coordinates=637877
```

Convert the streams to vector for better visualization

Inundation extent using HAND methodology

```
# compute the flow accumulation, drainage and streams (with threshold v
r.watershed elevation=elevation accumulation=flowacc \
 drainage=drainage stream=streams threshold=100000

# convert the streams to vector for better visualization
r.to.vect input=streams output=streams type=line

# compute new raster with elevation difference between
# the cell and the the cell on the stream where the cell drains
r.stream.distance stream_rast=streams direction=drainage \
 elevation=elevation method=downstream difference=above_stream

# compute a lake from specified coordinates and water level
r.lake elevation=elevation water_level=90 lake=lake coordinates=637877,

#display raster lake
d.mon wx0
d.rast elevation
```

Compute height difference between cell and cell on the stream

How does r.lake works?

```
#!/bin/bash
############################################################################
# Exercises for raster data processing in GRASS GIS
# GRASS GIS postgraduate course in Rio Cuarto
# Author: Veronica Andreo
# October, 2018
############################################################################



#
# Landscape analysis and forest fragmentation
#



# install addons
g.extension r.diversity
g.extension r.forestfrag
```

How does r.lake works?

```
# convert the streams to vector for better visualization
r.to.vect input=streams output=streams type=line

# compute new raster with elevation difference between
# the cell and the the cell on the stream where the cell drains
r.stream.distance stream_rast=streams direction=drainage \
 elevation=elevation method=downstream difference=above_stream

# compute a lake from specified coordinates and water level
r.lake elevation=elevation water_level=90 lake=lake coordinates=637877,

#display raster lake
d.mon wx0
d.rast elevation
d.rast lake

# simulate 5m inundation with HAND, specify stream as seed
```

Compute a lake from specified coordinates and water level

***Task**: Display the lake map over elevation map*

Inundation extent using HAND methodology

```
#!/bin/bash
############################################################################
# Exercises for raster data processing in GRASS GIS
# GRASS GIS postgraduate course in Rio Cuarto
# Author: Veronica Andreo
# October, 2018
############################################################################



#
# Landscape analysis and forest fragmentation
#



# install addons
g.extension r.diversity
g.extension r.forestfrag
```

Inundation extent using HAND methodology

```
# compute a lake from specified coordinates and water level
r.lake elevation=elevation water_level=90 lake=lake coordinates=637877,

#display raster lake
d.mon wx0
d.rast elevation
d.rast lake

# simulate 5m inundation with HAND, specify stream as seed
r.lake elevation=above_stream water_level=5 lake=flood seed=streams

# create a series of inundation maps with rising water levels
r.lake.series elevation=above_stream \
 start_water_level=0 end_water_level=5 \
 water_level_step=0.5 output=inundation \
 seed_raster=streams
```

Simulate 5-meter inundation from the streams

Inundation extent using HAND methodology

```
d.mon wx0
d.rast elevation
d.rast lake


# simulate 5m inundation with HAND, specify stream as seed
r.lake elevation=above_stream water_level=5 lake=flood seed=streams


# create a series of inundation maps with rising water levels
r.lake.series elevation=above_stream \
 start_water_level=0 end_water_level=5 \
 water_level_step=0.5 output=inundation \
 seed_raster=streams


# compute the volume and extent of flood water using t.rast.univar
t.rast.univar input=inundation separator=comma


# Visualize the inundation using the Animation Tool
g.gui.animation strds=inundation
```

Create a series of inundation maps with rising water level

Inundation extent using HAND methodology

```
# simulate 5m inundation with HAND, specify stream as seed
r.lake elevation=above_stream water_level=5 lake=flood seed=streams

# create a series of inundation maps with rising water levels
r.lake.series elevation=above_stream \
 start_water_level=0 end_water_level=5 \
 water_level_step=0.5 output=inundation \
 seed_raster=streams

# compute the volume and extent of flood water using t.rast.univar
t.rast.univar input=inundation separator=comma

# Visualize the inundation using the Animation Tool.
g.gui.animation strds=inundation


#
# Terrain analysis
```

Get volume and extent of flood for each time step

Inundation extent using HAND methodology

```
# create a series of inundation maps with rising water levels
r.lake.series elevation=above_stream \
 start_water_level=0 end_water_level=5 \
 water_level_step=0.5 output=inundation \
 seed_raster=streams

# compute the volume and extent of flood water using t.rast.univar
t.rast.univar input=inundation separator=comma

# Visualize the inundation using the Animation Tool.
g.gui.animation strds=inundation


#
# Terrain analysis
#
```

Create an animation with the output of r.lake.series

Create animation from GUI

- Launch it from menu File --> Animation tool
- *Add new animation* and click on *Add space-time dataset or series of map layers*
- Select *Space time raster dataset* and below select **inundation**
- Use *Add raster map layer* and select raster **elevation_shade** from PERMANENT
- Use *Add vector map layer* and select **streets_wake** from PERMANENT
- Select **inundation** layer and move it above elevation_shade
- Press OK and wait till the animation is rendered
- Press Play button

***Task**: Follow the instructions from the previous slide and create an animation with the output of r.lake.series*

Terrain analysis: r.geomorphon

Terrain analysis

```
#!/bin/bash
############################################################################
# Exercises for raster data processing in GRASS GIS
# GRASS GIS postgraduate course in Rio Cuarto
# Author: Veronica Andreo
# October, 2018
############################################################################



#
# Landscape analysis and forest fragmentation
#



# install addons
g.extension r.diversity
g.extension r.forestfrag
```

Terrain analysis

```
# Visualize the inundation using the Animation Tool.
g.gui.animation strds=inundation



#
# Terrain analysis
#



# set region
g.region raster=elevation

# get geomorphons
r.geomorphon elevation=elevation forms=elevation_geomorph

# extraction of summits
r.mapcalc expression="elevation_summits = if(elevation_geomorph == 2, 1
```

Set computational region

Terrain analysis

```
#
# Terrain analysis
#



# set region
g.region raster=elevation

# get geomorphons
r.geomorphon elevation=elevation forms=elevation_geomorph

# extraction of summits
r.mapcalc expression="elevation_summits = if(elevation_geomorph == 2, 1

# thining of summits raster and conversion to points
r.thin input=elevation_summits output=summits_thinned
r.to.vect input=summits_thinned output=summits type=point
```

Compute geo forms

Terrain analysis

```
#



# set region
g.region raster=elevation

# get geomorphons
r.geomorphon elevation=elevation forms=elevation_geomorph

# extraction of summits
r.mapcalc expression="elevation_summits = if(elevation_geomorph == 2, 1

# thining of summits raster and conversion to points
r.thin input=elevation_summits output=summits_thinned
r.to.vect input=summits_thinned output=summits type=point

# get height of summits
v.db.addcolumn map=summits columns="height double"
```

Extract summits with r.mapcalc

Terrain analysis

```
# set region
g.region raster=elevation

# get geomorphons
r.geomorphon elevation=elevation forms=elevation_geomorph

# extraction of summits
r.mapcalc expression="elevation_summits = if(elevation_geomorph == 2, 1

# thining of summits raster and conversion to points
r.thin input=elevation_summits output=summits_thinned
r.to.vect input=summits_thinned output=summits type=point

# get height of summits
v.db.addcolumn map=summits columns="height double"
v.what.rast map=summits raster=elevation column=height

# stats of summits
v.db.univar map=summits
```

Thin summits raster and convert to points

Terrain analysis

```
r.geomorphon elevation=elevation forms=elevation_geomorph

# extraction of summits
r.mapcalc expression="elevation_summits = if(elevation_geomorph == 2, 1

# thining of summits raster and conversion to points
r.thin input=elevation_summits output=summits_thinned
r.to.vect input=summits_thinned output=summits type=point

# get height of summits
v.db.addcolumn map=summits columns="height double"
v.what.rast map=summits raster=elevation column=height

# stats of summits
v.db.univar map=summits

# display output map
d.mon wx0
d.rast map=elevation_geomorph
```

Get summits' height

***Task**: Get summits height univariate statistics and display geomorphon map plus summits vector*

QUESTIONS?

Thanks for your attention!!

Move on to:

Satellite imagery processing in GRASS GIS

Presentation powered by

GitPitch