GRASS GIS 7:
A theoretical and practical course

Session 1 – Introduction to point sampling in G7

Markus Neteler

NINA 2016, Oslo, Norway

mundialis GmbH & Co. KG
http://www.mundialis.de

Session Objectives

- Random sampling of raster and vector points
- Stratified random sampling: Random sampling from vector map by attribute
- Stratified random sampling: Random sampling from vector map with spatial constraints
- Sampling at known positions: raster
- Sampling at known positions: vector
- GBIF example

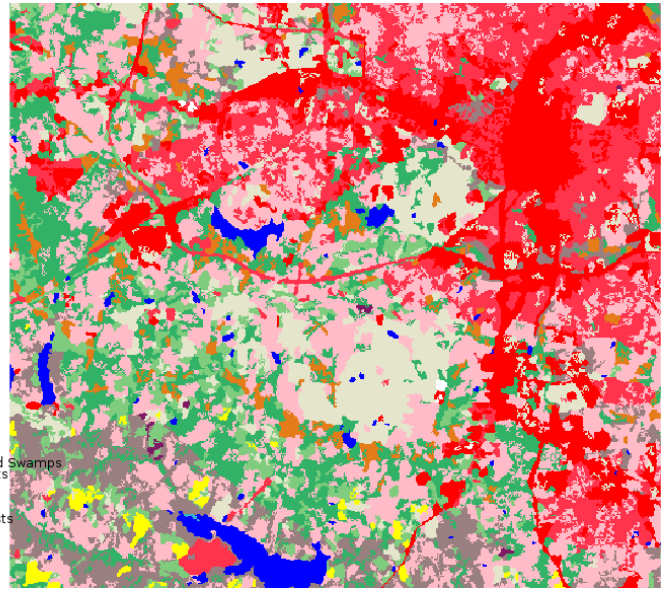
Exercise: Raster map random sampling

r.random - Creates a raster map layer and vector point map containing randomly located points.

Importantly, set the computational region first!

A) raster output

```
g.region raster=landuse96_28m -p

r.random input=landuse96_28m raster=lulc_100 n=100

# verify

r.category lulc_100


d.mon wx0

#note: small single raster cells are hard to see

d.rast lulc_100
```



Exercise: Raster map random sampling

B) vector output, extraction of collocated sampling data from two maps:

```
g.region raster=elevation -p

# random vector elevation points sampled from elevation map,
# with collocated values sampled from landuse map,
# result stored in 3D vector map:
r.random -d input=elevation cover=landuse96_28m vector=lulc_100 n=100

# data output (value: elevation, covervalue: landuse class):
v.db.select lulc_100

d.mon wx0
d.vect lulc_100

# verify 3D information:
v.info -g lulc_100
north=228365
south=215095
east=644805
west=630035
top=150.292648
bottom=62.751217
```

Exercise: Stratified random sampling

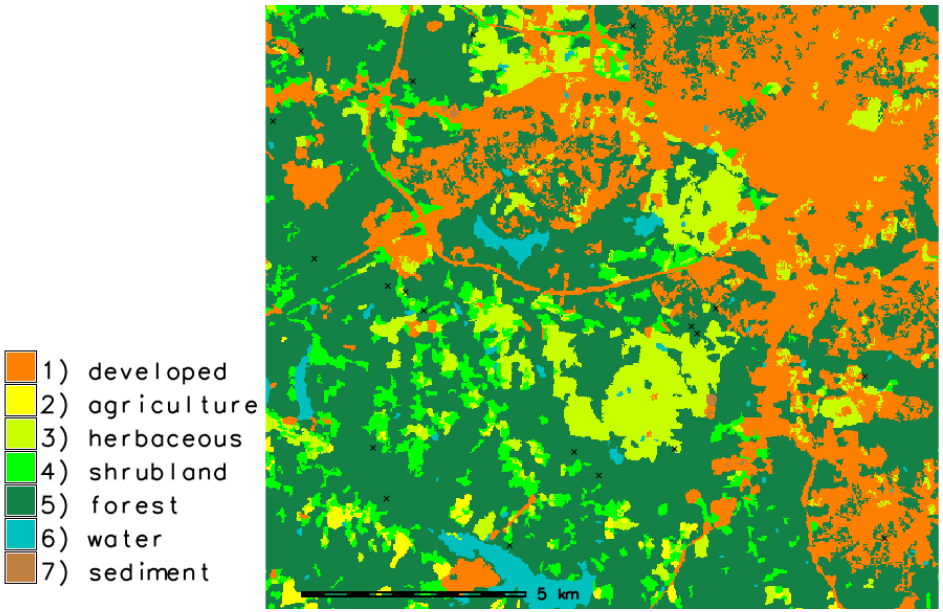
Constraining the random sampling to selected areas

see also: https://grass.osgeo.org/grass72/manuals/v.random.html#examples

Generate 20 random points restricted to forested areas:

```
g.region -p raster=landclass96

# -v Use raster values as categories instead of unique sequence (int only)
r.to.vect -v input=landclass96 output=landclass96 type=area

# use SQL query to restrict:
v.random restrict=landclass96 output=random_samples_forest npoints=20 \
  where="label = 'forest'" layer=1

d.mon wx0
d.rast landclass96
d.vect random_samples_forest
```

Using this flag also the number of samples can be controlled:
-a Generate n points for each individual area



Exercise: Sampling at known positions: raster

v.what.rast - Uploads raster values at positions of vector points to the table

Comparing elevation points: DEM versus geodetic heights

```
# we create a copy of the original geodetic points map in PERMANENT
# since we want to add a new column (map must be in own mapset)
g.copy vector=geodetic_pts,mygeodetic_pts

# set computational region to raster map to be queried
g.region raster=elev_state_500m -p

# query raster cells (a new column will be added to existing table)
v.what.rast map=mygeodetic_pts raster=elev_state_500m column=height

# see column list
v.info -c mygeodetic_pts

# compare official geodetic heights to those of elevation model
v.db.select map=mygeodetic_pts columns=Z_VALUE,height separator=comma

# univariate statistics
v.univar map=mygeodetic_pts column=height type=point
```

Exercise: Sampling at known positions: vector

v.what.vect - Uploads vector values at positions of vector points to the table

Uploading more vector attributes from a vector map

```
# we create a copy of the original hospital points map in PERMANENT
# since we want to add a new column (map must be in own mapset)
g.copy vector=hospitals,myhospitals

# add the new column (define type):
v.db.addcolumn myhospitals column="urb_name varchar(25)"

# query other vector map
v.what.vect myhospitals query_map=urbanarea column=urb_name query_column=NAME

# verify results:
v.db.select myhospitals columns=NAME,urb_name
NAME|urb_name
Cherry Hospital|
Wayne Memorial Hospital|
Watauga Medical Center|Boone
Central Prison Hospital|Raleigh
Dorothea Dix Hospital|Raleigh
Eastern Wake Hospital|Zebulon–Wendell
...
```

Further sampling methods

- v.sample - Samples a raster map at vector point locations, sampling interpolation method options: nearest, bilinear, bicubic

Sampling of temporal datasets

- t.rast.what - Sample a space time raster dataset at specific vector point coordinates and write the output to stdout using different layouts
- t.vect.observe.strds - Observes specific locations in a space time raster dataset over a period of time using vector points
- t.vect.what.strds - Stores raster map values at spatial and temporal positions of vector points as vector attributes
- v.what.strds - Uploads space time raster dataset values at positions of vector points to the table

Exercise: GBIF related env data sampling

GBIF.org: Free and Open Access to Biodiversity Data

http://www.gbif.org

Using the addon "v.in.pygbif" to import GBIF data into the North Carolina location

Picea rubens (Red Spruce): https://www.ibiblio.org/pic/NCTrees/redspruce.htm
Salix nigra (Black Willow): https://www.ibiblio.org/pic/NCTrees/blackwillow.htm

```
g.extension v.in.pygbif

g.region -p region=nc_500m
# show data without saving to map
# -t   Print result of taxon matching in table format and exit
v.in.pygbif -t output=nc_gbif taxa="Salix nigra,Picea rubens" rank=species

# -i   Produce individual map for each taxon
v.in.pygbif -i output=nc_gbif taxa="Salix nigra,Picea rubens" rank=species

# verify metadata
v.info Salix_nigra_nc_gbif
v.info Picea_rubens_nc_gbif
```

Exercise: GBIF related env data sampling

v.what.vect - Uploads vector values at positions of vector points to the table

```
# set computational region to NC but pixel resolution/alignment to geology
g.region -p region=nc_500m align=elev_state_500m

# for classified data direct query
# for continuous surface data interpolate from next 4 neighbors (-i)
v.what.rast -i map=Salix_nigra_nc_gbif raster=elev_state_500m column=elev500m
v.db.select Salix_nigra_nc_gbif

# anyone left out?
v.db.select Salix_nigra_nc_gbif where="elev500m IS NULL"
v.db.select Salix_nigra_nc_gbif where="elev500m IS NOT NULL"

v.univar Salix_nigra_nc_gbif column=elev500m type=point
```

Likewise BIOCLIM, MODIS LST, Sentinel-2 NDVI, MODIS snow presence, ... can be uploaded to the attribute table.