

Aplicații ale limbajului R în procesarea datelor geospațiale

Alexandru Dumitrescu

Administrația Națională de Meteorologie, Secția de Climatologie

"Soluții open source pentru prelucrarea și reprezentarea datelor
geospațiale", 19-20 aprilie 2013 - Cluj-Napoca



Introducere

I/O datelor geospațiale

Proiecții și transformări în sisteme de coordonate

Vizualizare

Prelucrare

Analiza de regresie

Metode de interpolare

Produse

Concluzii

Ce reprezintă limbajul R?

Ce reprezintă limbajul R?

un mediu relativ nou de analiză statistică și vizualizare a seturilor de date.

Ce reprezintă limbajul R?

un mediu relativ nou de analiză statistică și vizualizare a seturilor de date.

- ▶ a fost dezvoltat în anii '90 în Auckland, Noua Zeelandă, de către Ross Ihaka și Robert Gentleman;

Ce reprezintă limbajul R?

un mediu relativ nou de analiză statistică și vizualizare a seturilor de date.

- ▶ a fost dezvoltat în anii '90 în Auckland, Noua Zeelandă, de către Ross Ihaka și Robert Gentleman;
- ▶ fiind o implementare a limbajului S în mediul open-source, inițial limbajul R dispunea numai de tehnici de analiză statistică clasică;

Ce reprezintă limbajul R?

un mediu relativ nou de analiză statistică și vizualizare a seturilor de date.

- ▶ a fost dezvoltat în anii '90 în Auckland, Noua Zeelandă, de către Ross Ihaka și Robert Gentleman;
- ▶ fiind o implementare a limbajului S în mediul open-source, inițial limbajul R dispunea numai de tehnici de analiză statistică clasică;
- ▶ spre sfârșitul anilor '90 apar primele funcții de analiză statistică a seturilor de date spațiale (Venables and Ripley, 1999), acestea constituind primii pași făcuți în implementarea claselor de date spațiale în R.;

Ce reprezintă limbajul R?

un mediu relativ nou de analiză statistică și vizualizare a seturilor de date.

- ▶ a fost dezvoltat în anii '90 în Auckland, Noua Zeelandă, de către Ross Ihaka și Robert Gentleman;
- ▶ fiind o implementare a limbajului S în mediul open-source, inițial limbajul R dispunea numai de tehnici de analiză statistică clasică;
- ▶ spre sfârșitul anilor '90 apar primele funcții de analiză statistică a seturilor de date spațiale (Venables and Ripley, 1999), acestea constituind primii pași făcuți în implementarea claselor de date spațiale în R.;
- ▶ poate fi utilizat pe orice tip de platformă: Linux, MacOS X, Windows;

Ce reprezintă limbajul R?

un mediu relativ nou de analiză statistică și vizualizare a seturilor de date.

- ▶ a fost dezvoltat în anii '90 în Auckland, Noua Zeelandă, de către Ross Ihaka și Robert Gentleman;
- ▶ fiind o implementare a limbajului S în mediul open-source, inițial limbajul R dispunea numai de tehnici de analiză statistică clasică;
- ▶ spre sfârșitul anilor '90 apar primele funcții de analiză statistică a seturilor de date spațiale (Venables and Ripley, 1999), acestea constituind primii pași făcuți în implementarea claselor de date spațiale în R.;
- ▶ poate fi utilizat pe orice tip de platformă: Linux, MacOS X, Windows;
- ▶ dispune de un număr impresionant de pachete suplimentare (biblioteci) care completează sistemul de bază (în momentul de față sunt peste 100 de biblioteci disponibile în categoria *Spatial* <http://cran.r-project.org/web/views/Spatial.html>).

I/O datelor geospațiale

Biblioteci: **rgdal**, **raster**, **netcdf**.

Aplicații ale limbajului
R în procesarea datelor
geospațiale

Alexandru Dumitrescu

Introducere

I/O datelor geospațiale

Proiecții și transformări
în sisteme de
coordonate

Vizualizare

Prelucrare

Analiza de regresie

Metode de interpolare

Produse

Concluzii

Bibliografie

I/O datelor geospațiale

Biblioteci: **rgdal**, **raster**, **netcdf**.

- ▶ Cu ajutorul bibliotecii **rgdal** se pot citi majoritatea tipurilor de date geospațiale, de tip raster sau vector (puncte, linii sau poligoane):

Aplicații ale limbajului
R în procesarea datelor
geospațiale

Alexandru Dumitrescu

Introducere

I/O datelor geospațiale

Proiecții și transformări
în sisteme de
coordonate

Vizualizare

Prelucrare

Analiza de regresie

Metode de interpolare

Produse

Concluzii

Bibliografie

I/O datelor geospațiale

Biblioteci: **rgdal**, **raster**, **netcdf**.

- ▶ Cu ajutorul bibliotecii **rgdal** se pot citi majoritatea tipurilor de date geospațiale, de tip raster sau vector (puncte, linii sau poligoane):

```
>setwd("Documents/prezentari/
2013/geospatial_cluj/")
>library(rgdal)
>dem<-readGDAL("grids/dem.tif")
>summary(dem)
```

```
Object of class SpatialGridDataFrame
      min      max
x 133000 881000
y 225000 755000
Is projected: NA
proj4string : [NA]
Grid attributes:
  cellcentre.offset cellsize cells.dim
x              133500     1000       748
y              225500     1000       530
Data attributes:
      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
      0.0   87.0   175.0   310.5  409.0  2273.0
```


I/O datelor geospațiale

Biblioteci: **rgdal**, **raster**, **netcdf**.

- Cu ajutorul bibliotecii **rgdal** se pot citi majoritatea tipurilor de date geospațiale, de tip raster sau vector (puncte, linii sau poligoane):

```
Object of class SpatialGridDataFrame
      min      max
x 133000 881000
y 225000 755000
Is projected: NA
proj4string : [NA]
Grid attributes:
      cellcentre.offset cellsize cells.dim
x          133500         1000         748
y          225500         1000         530
Data attributes:
      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
      0.0   87.0   175.0   310.5  409.0  2273.0

Object of class SpatialPointsDataFrame
      min      max
coords.x1 20.60157 29.75894
coords.x2 43.66152 48.19488
Is projected: FALSE
proj4string :
[+proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0]
Number of points: 164
Data attributes:
      JU      CODST      CODGE      NUME
CS : 9 Min. :15000 Min. :340521 ADAMCLISI : 1
TL : 8 1st Qu.:15142 1st Qu.:453121 ADJUD : 1
BH : 7 Median :15278 Median :537520 ALBA IULIA: 1
CJ : 7 Mean :15263 Mean :553963 ALEXANDRIA: 1
CT : 7 3rd Qu.:15376 3rd Qu.:639287 ARAD : 1
AB : 5 Max. :15499 Max. :812637 BACAU : 1
(Other):121 (Other) :158
```

```
>setwd("Documents/prezentari/
2013/geospatial_cluj/")
>library(rgdal)
>dem<-readGDAL("grids/dem.tif")
>summary(dem)

>statii<-readOGR("shp",
"statii_meteo_4326")
>summary(statii)
```

- Biblioteca **raster**, prin funcția *stack*, poate crea un obiect de tip raster multiband din mai multe fișiere de tip raster, chiar dacă formatul acestora este diferit:

- Biblioteca **raster**, prin funcția *stack*, poate crea un obiect de tip raster multiband din mai multe fișiere de tip raster, chiar dacă formatul acestora este diferit:

```
[1] "grids/dem.tif" "grids/lat.tif" "grids/lon.tif"
"grids/TWI.asc"

class      : RasterBrick
dimensions : 530, 748, 396440, 4  (nrow, ncol, ncell, nlayers)
resolution : 1000, 1000  (x, y)
extent     : 133000, 881000, 225000, 755000  (xmin, xmax,
ymain, ymax)
coord. ref. : NA
data source : in memory
names      :      dem,          lat,          lon,          TWI
min values  :    0.00000,    43.43079,    20.06989,    15.31612
max values  :  2273.00000,    48.28944,    30.11780,    25.03275
```

- Fișierele multidimensionale de tip *.nc pot fi aduse în mediul R utilizându-se biblioteca **ncdf**. Funcțiile de bază ale acestei biblioteci sunt *open.ncdf* și *get.var.ncdf*.

- Fișierele multidimensionale de tip *.nc pot fi aduse în mediul R utilizându-se biblioteca **ncdf**. Funcțiile de bază ale acestei biblioteci sunt *open.ncdf* și *get.var.ncdf*.

```
>library(ncdf)
>nc <- open.ncdf("UCLM/UCLM-
PROMES_A1B_HadCM3Q0_25km
_MM-CRU_2011-2020_pr.nc")
>print(nc)

[1] "file UCLM/UCLM-PROMES_A1B_HadCM3Q0
_25km_MM-CRU_2011-2020_pr.nc has 3 dimensions:"
[1] "lon    Size: 278"
[1] "lat    Size: 170"
[1] "time   Size: 120"
[1] "-----"
[1] "file UCLM/UCLM-PROMES_A1B_HadCM3Q0
_25km_MM-CRU_2011-2020_pr.nc has 1 variables:"
[1] "float pr[lon,lat,time] Longname:
Precipitacion Missval:-9.99999979021477e+33"

>field <- get.var.ncdf(nc, "pr")
>dim(field)
[1] 278 170 120
>class(field)
[1] "array"
```

- Fișierele multidimensionale de tip *.nc pot fi aduse în mediul R utilizându-se biblioteca **ncdf**. Funcțiile de bază ale acestei biblioteci sunt *open.ncdf* și *get.var.ncdf*.

```
>library(ncdf)
>nc <- open.ncdf("UCLM/UCLM-
PROMES_A1B_HadCM3Q0_25km
_MM-CRU_2011-2020_pr.nc")
>print(nc)

[1] "file UCLM/UCLM-PROMES_A1B_HadCM3Q0
_25km_MM-CRU_2011-2020_pr.nc has 3 dimensions:"
[1] "lon    Size: 278"
[1] "lat    Size: 170"
[1] "time   Size: 120"
[1] "-----"
[1] "file UCLM/UCLM-PROMES_A1B_HadCM3Q0
_25km_MM-CRU_2011-2020_pr.nc has 1 variables:"
[1] "float pr[lon,lat,time] Longname:
Precipitacion Missval:-9.99999979021477e+33"

>field <- get.var.ncdf(nc, "pr")
>dim(field)
[1] 278 170 120
>class(field)
[1] "array"
```

- Salvarea datelor geospațiale din R se realizează cu ajutorul funcțiilor **rgdal** *writeGDAL*, pentru obiecte de tip raster, și *writeOGR* pentru, obiectele de tip vector:

```
>writeGDAL(dem,"grids/elevation.asc")
```

Proiecții și transformări în sisteme de coordonate

Biblioteci: **rgdal**, **raster**.

Aplicații ale limbajului
R în procesarea datelor
geospațiale

Alexandru Dumitrescu

Introducere

I/O datelor geospațiale

**Proiecții și transformări
în sisteme de
coordonate**

Vizualizare

Prelucrare

Analiza de regresie

Metode de interpolare

Produse

Concluzii

Bibliografie

Proiecții și transformări în sisteme de coordonate

Biblioteci: **rgdal**, **raster**.

- ✓ Transformare coordonatelor obiectelor de tip raster și vector se poate realiza în R prin utilizarea codurilor EPSG.

Aplicații ale limbajului
R în procesarea datelor
geospațiale

Alexandru Dumitrescu

Introducere

I/O datelor geospațiale

Proiecții și transformări
în sisteme de
coordonate

Vizualizare

Prelucrare

Analiza de regresie

Metode de interpolare

Produse

Concluzii

Bibliografie

Proiecții și transformări în sisteme de coordonate

Biblioteci: **rgdal**, **raster**.

- ✓ Transformare coordonatelor obiectelor de tip raster și vector se poate realiza în R prin utilizarea codurilor EPSG.
- ✓ Identificare și definirea sistemului de coordonate realizează prin funcțiile *proj4string* și *projection*.

Proiecții și transformări în sisteme de coordonate

Biblioteci: **rgdal**, **raster**.

- ✓ Transformare coordonatelor obiectelor de tip raster și vector se poate realiza în R prin utilizarea codurilor EPSG.
- ✓ Identificare și definirea sistemului de coordonate realizează prin funcțiile *proj4string* și *projection*.
- ✓ Transformarea coordonatelor presupune utilizarea funcțiilor *spTransform*, pentru obiecte de tip vector, și *projectRaster*, pentru obiecte de tip raster:

Biblioteci: **rgdal**, **raster**.

- ✓ Transformare coordonatelor obiectelor de tip raster și vector se poate realiza în R prin utilizarea codurilor EPSG.
- ✓ Identificare și definirea sistemului de coordonate realizează prin funcțiile *proj4string* și *projection*.
- ✓ Transformarea coordonatelor presupune utilizarea funcțiilor *spTransform*, pentru obiecte de tip vector, și *projectRaster*, pentru obiecte de tip raster:

```
>statii<-spTransform(statii,CRS("+init=epsg:3844"))
> proj4string(statii)
[1] "+init=epsg:3844 +proj=sterea +lat_0=46 +lon_0=25 +k=0.99975 +x_0=500000 +y_0=500000
+ellps=krass +tows84=33.4,-146.6,-76.3,-0.359,-0.053,0.844,-0.84 +units=m +no_defs"

>dem<-raster("grids/dem.tif")
>projection(dem)
[1] "NA"
>projection(dem)<-"+init=epsg:3844"
>dem<-projectRaster(dem,crs="+init=epsg:4326")
>projection(dem)
[1] "+init=epsg:4326 +proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +tows84=0,0,0"
>?projectRaster
```

Biblioteci: **sp**, **raster**, **rasterVis**, **lattice**, **RColorBrewer**.

Biblioteci: **sp**, **raster**, **rasterVis**, **lattice**, **RColorBrewer**.

- ✓ Reprezentarea grafică a datelor geospațiale se poate realiza în R, în majoritatea cazurilor, cu ajutorul funcțiilor *splot* și *plot*.

Biblioteci: **sp**, **raster**, **rasterVis**, **lattice**, **RColorBrewer**.

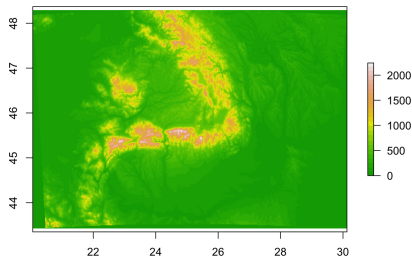
- ✓ Reprezentarea grafica a datelor geospațiale se poate realiza în R, în majoritatea cazurilor, cu ajutorul funcțiilor *splot* și *plot*.
- ✓ Biblioteca **RColorBrewer** ajută la întocmirea paletelor de culori, necesare în principal în reprezentarea sugestivă a hărților care au la bază informație de tip raster sau de tip vector - contururi umplute (<http://colorbrewer2.org/>).

Biblioteci: **sp**, **raster**, **rasterVis**, **lattice**, **RColorBrewer**.

- ✓ Reprezentarea grafică a datelor geospațiale se poate realiza în R, în majoritatea cazurilor, cu ajutorul funcțiilor *splot* și *plot*.
- ✓ Biblioteca **RColorBrewer** ajută la întocmirea paletelor de culori, necesare în principal în reprezentarea sugestivă a hărților care au la bază informație de tip raster sau de tip vector - contururi umplute (<http://colorbrewer2.org/>).
- ✓ Rutina de plotare *splot* se bazează pe funcția *xyplot* din biblioteca **lattice**, iar funcția *plot* se bazează pe funcția cu același nume din pachetul de bază R, fiind adaptată în cadrul bibliotecii **raster** pentru vizualizarea datelor geografice.

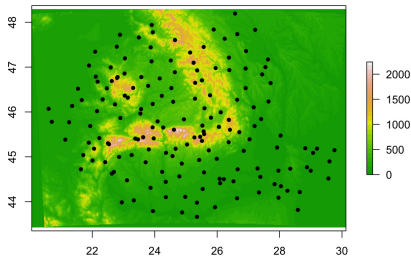
Vizualizare

```
>plot(dem,col=terrain.colors(25))
```



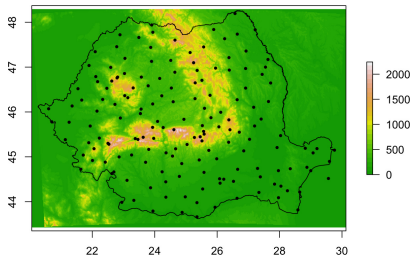
Vizualizare

```
>plot(dem,col=terrain.colors(25))  
>plot(statii, pch=19,add=T,cex=0.5)
```

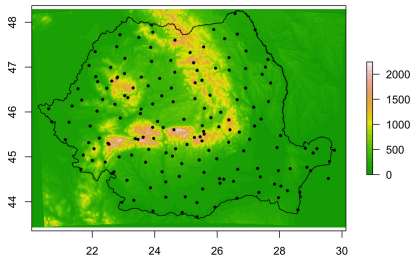


Vizualizare

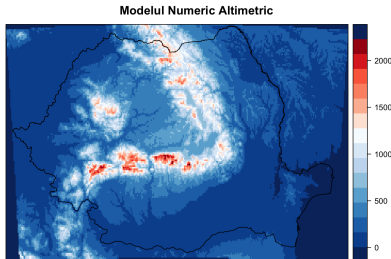
```
>plot(dem,col=terrain.colors(25))  
>plot(statii, pch=19,add=T,cex=0.5)  
>plot(granite,add=T)
```



```
>plot(dem,col=terrain.colors(25))  
  
>plot(statii, pch=19,add=T,cex=0.5)  
  
>plot(granite,add=T)
```



```
>library(RColorBrewer)  
>cols <- append(rev(brewer.pal(9,"Blues")),  
  brewer.pal(6,"Reds"))  
>gran<-list('sp.polygons',granite,  
  col='yellow',first=FALSE)  
>spplot(dem,col.regions=cols,  
  colorkey=list(space='right',title="metri"),  
  sp.layout=list(gran))
```



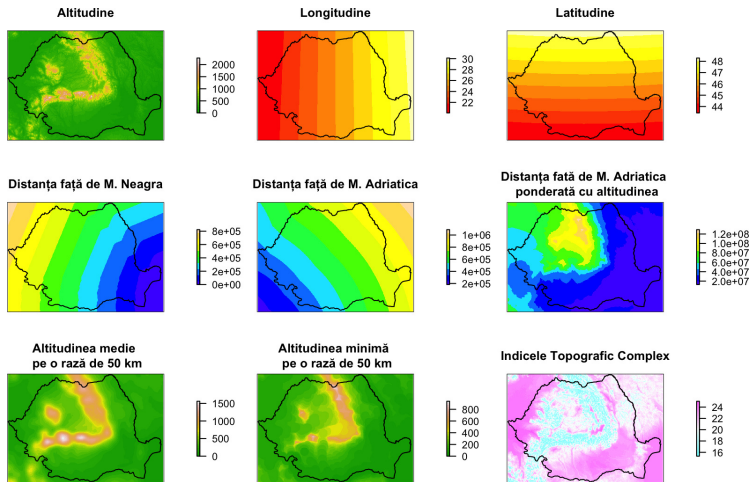


Figura: Vizualizarea simultană a mai multor hărți în R prin utilizarea funcției `plot` (predictori utilizați în spațializarea variabilelor climatice, derivați din Modelul Numeric Altimetric)

Biblioteci: **sp**, **raster**, **rgeos**.

Biblioteci: **sp**, **raster**, **rgeos**.

- ✓ Procesarea seturilor mari de date geospațiale este condiționată de capacitatea memoriei RAM a sistemului de calcul.

Biblioteci: **sp**, **raster**, **rgeos**.

- ✓ Procesarea seturilor mari de date geospațiale este condiționată de capacitatea memoriei RAM a sistemului de calcul.
- ✓ Pentru datele raster există posibilitatea citirii în R doar a anumitor bucăți din fișier cu funcția **rgdal** *getRasterData*;

Biblioteci: **sp**, **raster**, **rgeos**.

- ✓ Procesarea seturilor mari de date geospațiale este condiționată de capacitatea memoriei RAM a sistemului de calcul.
 - ✓ Pentru datele raster există posibilitatea citirii în R doar a anumitor bucăți din fișier cu funcția **rgdal** *getRasterData*;
 - ✓ Pentru fișierele raster foarte mari există posibilitatea stocării informație pe disk cu ajutorul funcției *fromDisk* din **raster**, acestea fiind procesate în R bucată cu bucată (in chunks)

Biblioteci: **sp**, **raster**, **rgeos**.

- ✓ Procesarea seturilor mari de date geospațiale este condiționată de capacitatea memoriei RAM a sistemului de calcul.
 - ✓ Pentru datele raster există posibilitatea citirii în R doar a anumitor bucăți din fișier cu funcția **rgdal** *getRasterData*;
 - ✓ Pentru fișierele raster foarte mari există posibilitatea stocării informație pe disk cu ajutorul funcției *fromDisk* din **raster**, acestea fiind procesate în R bucată cu bucată (in chunks)
- ✓ Sintaxa funcțiilor aplicate datelor geospațiale de tip vector este încă destul de dificilă.

Biblioteci: **sp**, **raster**, **rgeos**.

- ✓ Procesarea seturilor mari de date geospațiale este condiționată de capacitatea memoriei RAM a sistemului de calcul.
 - ✓ Pentru datele raster există posibilitatea citirii în R doar a anumitor bucăți din fișier cu funcția **rgdal** *getRasterData*;
 - ✓ Pentru fișierele raster foarte mari există posibilitatea stocării informație pe disk cu ajutorul funcției *fromDisk* din **raster**, acestea fiind procesate în R bucată cu bucată (in chunks)
- ✓ Sintaxa funcțiilor aplicate datelor geospațiale de tip vector este încă destul de dificilă.
- ✓ R poate apela funcțiile de procesare ale altor aplicații SIG prin intermediul bibliotecilor: **RSAGA** (SAGA-GIS), **spgrass6** (GRASS), **RPyGeo** (Geoprocessing Tools din ArcGIS), **spsextante** (Sextante).

- Decuparea datelor de tip raster

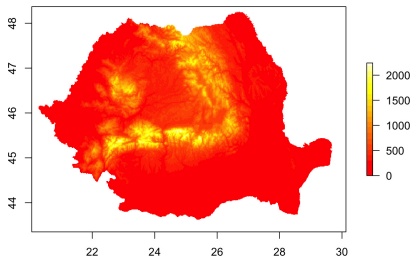
- ▶ Decuparea datelor de tip raster
 - ▶ mască vectorială

- ▶ Decuparea datelor de tip raster
 - ▶ mască vectorială

```
>r1 <- rasterize(granite, dem,  
mask=TRUE, progress='text')
```

- Decuparea datelor de tip raster
 - mască vectorială

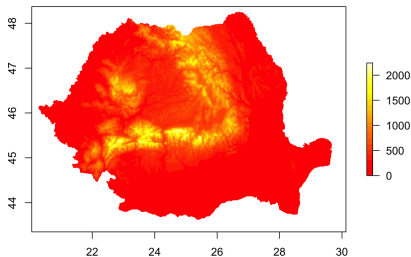
```
>r1 <- rasterize(granite, dem,  
mask=TRUE, progress='text')  
>plot(r1,col=heat.colors(21))
```



► Decuparea datelor de tip raster

► mască vectorială

```
>r1 <- rasterize(granite, dem,  
mask=TRUE, progress='text')  
>plot(r1,col=heat.colors(21))
```

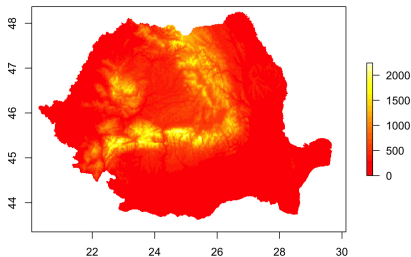


► dreptunghi delimitat de coordonatele geografice

► Decuparea datelor de tip raster

► mască vectorială

```
>r1 <- rasterize(granite, dem,  
mask=TRUE, progress='text')  
>plot(r1,col=heat.colors(21))
```



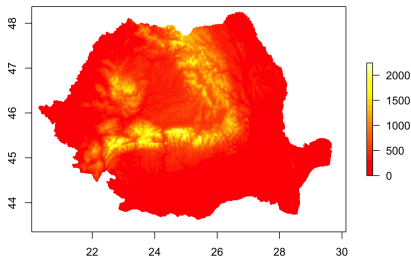
► dreptunghi delimitat de coordonatele geografice

```
>limita <- c(22, 27, 45, 48)
```

► Decuparea datelor de tip raster

► mască vectorială

```
>r1 <- rasterize(granite, dem,  
mask=TRUE, progress='text')  
>plot(r1,col=heat.colors(21))
```



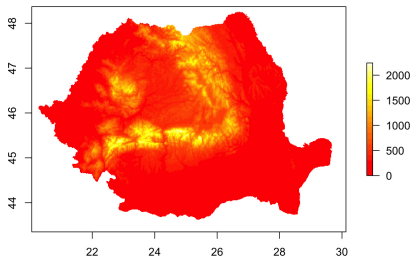
► dreptunghi delimitat de coordonatele geografice

```
>limita <- c(22, 27, 45, 48)  
>r2 <- crop(dem, limita)
```

► Decuparea datelor de tip raster

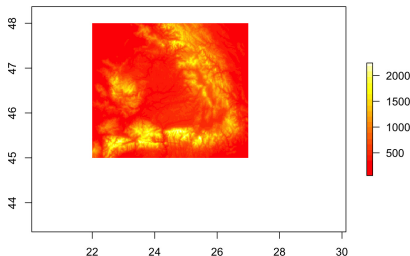
► mască vectorială

```
>r1 <- rasterize(granite, dem,  
mask=TRUE, progress='text')  
>plot(r1,col=heat.colors(21))
```



► dreptunghi delimitat de coordonatele geografice

```
>limita <- c(22, 27, 45, 48)  
>r2 <- crop(dem, limita)  
>plot(r2,col=heat.colors(21))
```



► Extragerea elementelor morfometrice

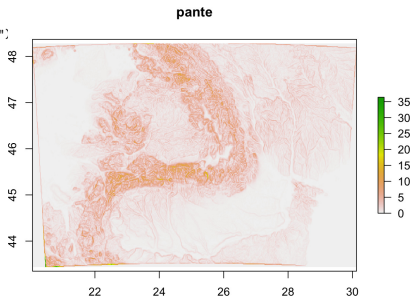
► Extragerea elementelor morfometrice

```
>r3 <- terrain(dem, opt = c("slope", "aspect"),  
unit = "degrees")
```

► Extragerea elementelor morfometrice

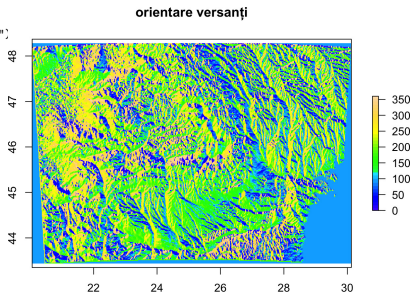
```
>r3 <- terrain(dem, opt = c("slope", "aspect"),  
unit = "degrees")
```

```
>plot(r3[[1]],main="pante")
```



► Extragerea elementelor morfometrice

```
>r3 <- terrain(dem, opt = c("slope", "aspect"),  
unit = "degrees")  
  
>plot(r3[[1]],main="pante")  
  
>plot(r3[[2]],main="orientare versanți",  
col=topo.colors(36))
```



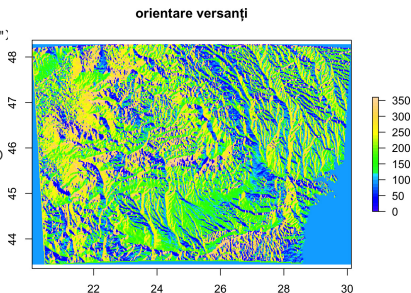
► Extragerea elementelor morfometrice

```
>r3 <- terrain(dem, opt = c("slope", "aspect"),  
unit = "degrees")
```

```
>plot(r3[[1]],main="pante")
```

```
>plot(r3[[2]],main="orientare versanți",  
col=topo.colors(36))
```

```
>hill <- hillShade(r3[[1]], r3[[2]], 40, 270)
```



► Extragerea elementelor morfometrice

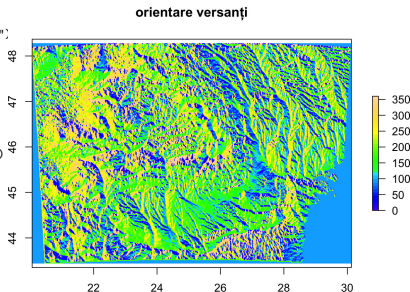
```
>r3 <- terrain(dem, opt = c("slope", "aspect"),  
unit = "degrees")
```

```
>plot(r3[[1]],main="pante")
```

```
>plot(r3[[2]],main="orientare versanți",  
col=topo.colors(36))
```

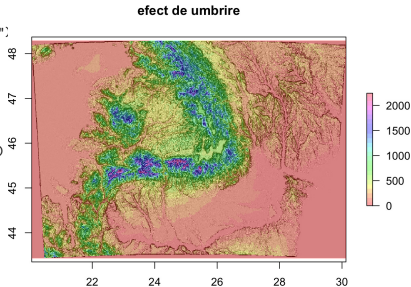
```
>hill <- hillShade(r3[[1]], r3[[2]], 40, 270)
```

```
>plot(hill, col = grey(0:100/100),  
legend = FALSE, main = "efect de umbrire")
```



► Extragerea elementelor morfometrice

```
>r3 <- terrain(dem, opt = c("slope", "aspect"),  
unit = "degrees")  
  
>plot(r3[[1]],main="pante")  
  
>plot(r3[[2]],main="orientare versanți",  
col=topo.colors(36))  
  
>hill <- hillShade(r3[[1]], r3[[2]], 40, 270)  
  
>plot(hill, col = grey(0:100/100),  
legend = FALSE, main = "efect de umbrire")  
  
>plot(dem, col = rainbow(25, alpha = 0.35),  
add = TRUE)
```



► Extragerea elementelor morfometrice

```
>r3 <- terrain(dem, opt = c("slope", "aspect"),  
unit = "degrees")
```

```
>plot(r3[[1]],main="pante")
```

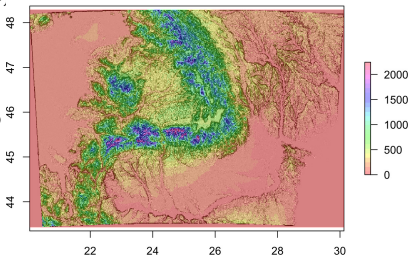
```
>plot(r3[[2]],main="orientare versanți",  
col=topo.colors(36))
```

```
>hill <- hillShade(r3[[1]], r3[[2]], 40, 270)
```

```
>plot(hill, col = grey(0:100/100),  
legend = FALSE, main = "efect de umbrire")
```

```
>plot(dem, col = rainbow(25, alpha = 0.35),  
add = TRUE)
```

efect de umbrire



► Raster calculator

► Extragerea elementelor morfometrice

```
>r3 <- terrain(dem, opt = c("slope", "aspect"),  
unit = "degrees")
```

```
>plot(r3[[1]],main="pante")
```

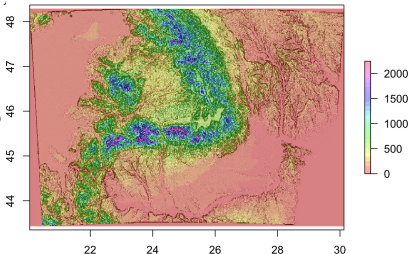
```
>plot(r3[[2]],main="orientare versanți",  
col=topo.colors(36))
```

```
>hill <- hillShade(r3[[1]], r3[[2]], 40, 270)
```

```
>plot(hill, col = grey(0:100/100),  
legend = FALSE, main = "efect de umbrire")
```

```
>plot(dem, col = rainbow(25, alpha = 0.35),  
add = TRUE)
```

efect de umbrire



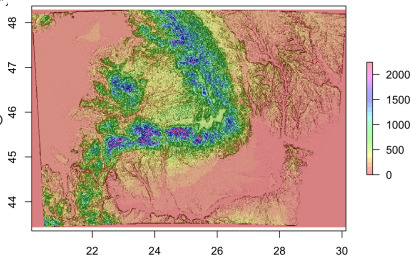
► Raster calculator

```
>dem2<-dem/100
```

► Extragerea elementelor morfometrice

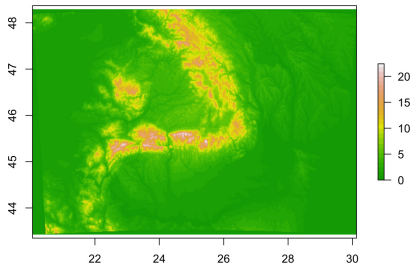
```
>r3 <- terrain(dem, opt = c("slope", "aspect"),  
unit = "degrees")  
  
>plot(r3[[1]],main="pante")  
  
>plot(r3[[2]],main="orientare versanți",  
col=topo.colors(36))  
  
>hill <- hillShade(r3[[1]], r3[[2]], 40, 270)  
  
>plot(hill, col = grey(0:100/100),  
legend = FALSE, main = "efect de umbrire")  
  
>plot(dem, col = rainbow(25, alpha = 0.35),  
add = TRUE)
```

efect de umbrire



► Raster calculator

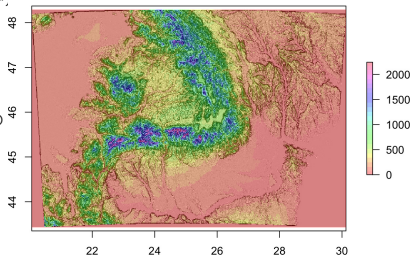
```
>dem2<-dem/100  
  
>plot(dem2,col=terrain.colors(25))
```



► Extragerea elementelor morfometrice

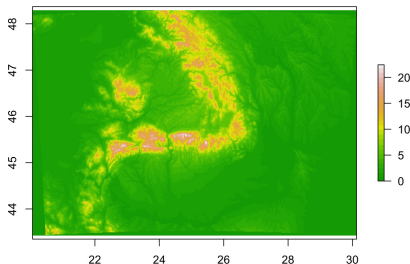
```
>r3 <- terrain(dem, opt = c("slope", "aspect"),  
unit = "degrees")  
  
>plot(r3[[1]],main="pante")  
  
>plot(r3[[2]],main="orientare versanți",  
col=topo.colors(36))  
  
>hill <- hillShade(r3[[1]], r3[[2]], 40, 270)  
  
>plot(hill, col = grey(0:100/100),  
legend = FALSE, main = "efect de umbrire")  
  
>plot(dem, col = rainbow(25, alpha = 0.35),  
add = TRUE)
```

efect de umbrire



► Raster calculator

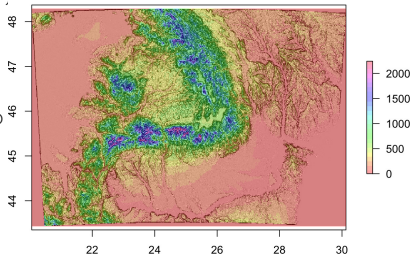
```
>dem2<-dem/100  
  
>plot(dem2,col=terrain.colors(25))  
  
>lonlat<-lon+lat
```



► Extragerea elementelor morfometrice

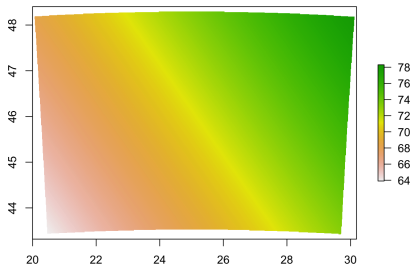
```
>r3 <- terrain(dem, opt = c("slope", "aspect"),  
unit = "degrees")  
  
>plot(r3[[1]],main="pante")  
  
>plot(r3[[2]],main="orientare versanți",  
col=topo.colors(36))  
  
>hill <- hillShade(r3[[1]], r3[[2]], 40, 270)  
  
>plot(hill, col = grey(0:100/100),  
legend = FALSE, main = "efect de umbrire")  
  
>plot(dem, col = rainbow(25, alpha = 0.35),  
add = TRUE)
```

efect de umbrire



► Raster calculator

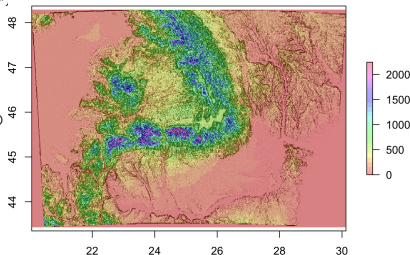
```
>dem2<-dem/100  
  
>plot(dem2,col=terrain.colors(25))  
  
>lonlat<-lon+lat  
  
>plot(lonlat)
```



► Extragerea elementelor morfometrice

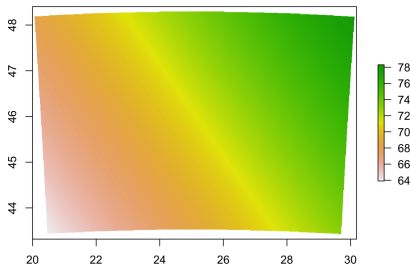
```
>r3 <- terrain(dem, opt = c("slope", "aspect"),  
unit = "degrees")  
  
>plot(r3[[1]],main="pante")  
  
>plot(r3[[2]],main="orientare versanți",  
col=topo.colors(36))  
  
>hill <- hillShade(r3[[1]], r3[[2]], 40, 270)  
  
>plot(hill, col = grey(0:100/100),  
legend = FALSE, main = "efect de umbrire")  
  
>plot(dem, col = rainbow(25, alpha = 0.35),  
add = TRUE)
```

efect de umbrire



► Raster calculator

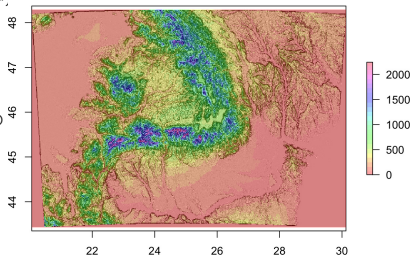
```
>dem2<-dem/100  
  
>plot(dem2,col=terrain.colors(25))  
  
>lonlat<-lon+lat  
  
>plot(lonlat)  
  
>lat[dem>1000]<-50
```



► Extragerea elementelor morfometrice

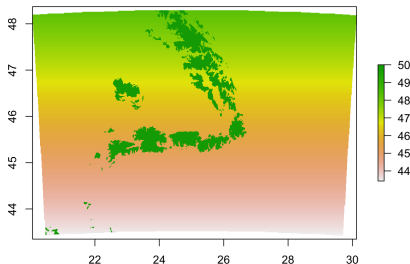
```
>r3 <- terrain(dem, opt = c("slope", "aspect"),  
unit = "degrees")  
  
>plot(r3[[1]],main="pante")  
  
>plot(r3[[2]],main="orientare versanți",  
col=topo.colors(36))  
  
>hill <- hillShade(r3[[1]], r3[[2]], 40, 270)  
  
>plot(hill, col = grey(0:100/100),  
legend = FALSE, main = "efect de umbrire")  
  
>plot(dem, col = rainbow(25, alpha = 0.35),  
add = TRUE)
```

efect de umbrire



► Raster calculator

```
>dem2<-dem/100  
  
>plot(dem2,col=terrain.colors(25))  
  
>lonlat<-lon+lat  
  
>plot(lonlat)  
  
>lat[dem>1000]<-50  
  
>plot(lat)
```



Analiza de regresie

Biblioteci: raster, sp, spgwr, mgcv

Aplicații ale limbajului
R în procesarea datelor
geospațiale

Alexandru Dumitrescu

Introducere

I/O datelor geospațiale

Proiecții și transformări
în sisteme de
coordonate

Vizualizare

Prelucrare

Analiza de regresie

Metode de interpolare

Produse

Concluzii

Bibliografie

Analiza de regresie

Biblioteci: **raster**, **sp**, **spgwr**, **mgcv**

- ✓ În R sunt implementate majoritatea metodelor de regresie

Analiza de regresie

Biblioteci: **raster**, **sp**, **spgwr**, **mgcv**

- ✓ În R sunt implementate majoritatea metodelor de regresie
- ✓ Pot fi aplicate în spațializarea atât metode de regresii parametrice (liniare), cât și semi-parametrice (regresia geografic ponderată) sau ne-parametrice (modelul aditiv generalizat)

Analiza de regresie

Biblioteci: **raster**, **sp**, **spgwr**, **mgcv**

- ✓ În R sunt implementate majoritatea metodelor de regresie
- ✓ Pot fi aplicate în spațializarea atât metode de regresii parametrice (liniare), cât și semi-parametrice (regresia geografic ponderată) sau ne-parametrice (modelul aditiv generalizat)
- ▶ Regresia liniară (Thom, 1966)

Biblioteci: **raster**, **sp**, **spgwr**, **mgcv**

- ✓ În R sunt implementate majoritatea metodelor de regresie
 - ✓ Pot fi aplicate în spațializarea atât metode de regresii parametrice (liniare), cât și semi-parametrice (regresia geografic ponderată) sau ne-parametrice (modelul aditiv generalizat)
- Regresia liniară (Thom, 1966)

```
>statii<-readOGR("shp","statii_meteo_4326")
>temp<-read.csv('http://earth.unibuc.ro/file_
download/27415/tt_01_2005.csv', na.string="-")

>temp.co<-merge(statii,temp,
by.x="CODGE",by.y="cod")
>coordinates(temp.co)<--coords.x1 + coords.x2
>ex<-extract(rs,temp.co)
>temp.co$dem<-ex[, 'dem']
>temp.co$lon<-ex[, 'lon']
>temp.co$lat<-ex[, 'lat']
>temp.co$TWI<-ex[, 'TWI']

>lm1<-lm(temp_med1-dem+lon+lat+TWI,temp.co)
>summary(lm1)$r.squared
[1] 0.915
>rtt<-predict(rs,lm1)
```

Analiza de regresie

Biblioteci: **raster**, **sp**, **spgwr**, **mgcv**

- ✓ În R sunt implementate majoritatea metodelor de regresie
- ✓ Pot fi aplicate în spațializarea atât metode de regresii parametrice (liniare), cât și semi-parametrice (regresia geografic ponderată) sau ne-parametrice (modelul aditiv generalizat)

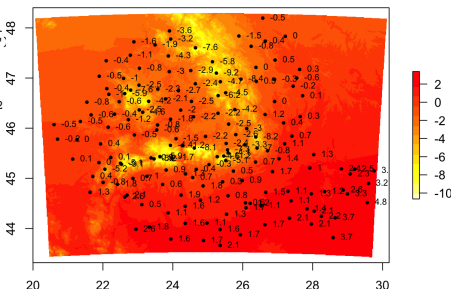
► Regresia liniară (Thom, 1966)

```
>statii<-readOGR("shp","statii_meteo_4326")
>temp<-read.csv('http://earth.unibuc.ro/file_48
download/27415/tt_01_2005.csv', na.string="-")
```

```
>temp.co<-merge(statii,temp,
by.x="CODGE",by.y="cod")
>coordinates(temp.co)<--coords.x1 + coords.x2
>ex<-extract(rs,temp.co)
>temp.co$dem<-ex[, 'dem']
>temp.co$lon<-ex[, 'lon']
>temp.co$lat<-ex[, 'lat']
>temp.co$TWI<-ex[, 'TWI']
```

```
>lm1<-lm(temp_med1-dem+lon+lat+TWI,temp.co)
>summary(lm1)$r.squared
[1] 0.915
>rtt<-predict(rs,lm1)
```

```
>plot(rtt,col=rev(heat.colors(14)))
>plot(temp.co,add=T,pch=19,cex=0.6)
>text(coordinates(temp.co)[,1],
coordinates(temp.co)[,2],temp.co$temp_med1,cex=0.7,pos=4)
```



Analiza de regresie

- Regresia geografic ponderată (Fotheringham et al., 2002)

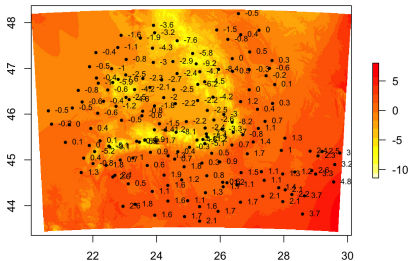
► Regresia geografic ponderată (Fotheringham et al., 2002)

```
>library(spgwr)
>rsp<-as(rs,"SpatialPixelsDataFrame")
>gwr.band <- gwr.sel(formula(lm1), data=temp.co)
>gwr.fit <- gwr(formula(lm1), temp.co,
bandwidth=gwr.band)
>gwr.r <- gwr(formula(lm1), temp.co,
bandwidth=gwr.band,
fit.points = rsp, predict=TRUE)
rgwr<-raster(gwr.r$SDF['GWR'])
```

► Regresia geografic ponderată (Fotheringham et al., 2002)

```
>library(spgwr)
>rsp<-as(rs,"SpatialPixelsDataFrame")
>gwr.band <- gwr.sel(formula(lm1), data=temp.co)
>gwr.fit <- gwr(formula(lm1), temp.co,
bandwidth=gwr.band)
>gwr.r <- gwr(formula(lm1), temp.co,
bandwidth=gwr.band,
fit.points = rsp, predict=TRUE)
rgwr<-raster(gwr.r$SDF['GWR'])

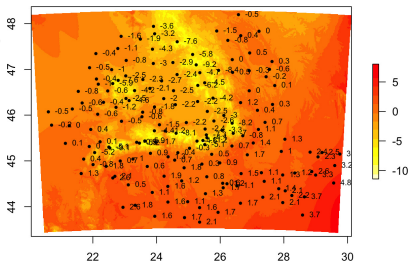
>plot(rgwr,col=rev(heat.colors(14)))
>plot(temp.co,add=T,pch=19,cex=0.5)
>text(coordinates(temp.co)[,1],
coordinates(temp.co)[,2],
temp.co$temp_med1,cex=0.7,pos=4)
```



► Regresia geografic ponderată (Fotheringham et al., 2002)

```
>library(spgwr)
>rsp<-as(rs,"SpatialPixelsDataFrame")
>gwr.band <- gwr.sel(formula(lm1), data=temp.co)
>gwr.fit <- gwr(formula(lm1), temp.co,
bandwidth=gwr.band)
>gwr.r <- gwr(formula(lm1), temp.co,
bandwidth=gwr.band,
fit.points = rsp, predict=TRUE)
rgwr<-raster(gwr.r$SDF['GWR'])

>plot(rgwr,col=rev(heat.colors(14)))
>plot(temp.co,add=T,pch=19,cex=0.5)
>text(coordinates(temp.co)[,1],
coordinates(temp.co)[,2],
temp.co$temp_med1,cex=0.7,pos=4)
```



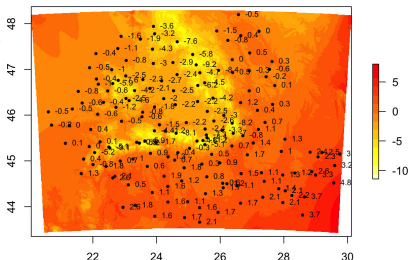
► Modelul aditiv generalizat (Wood and Augustin, 2002)

Analiza de regresie

► Regresia geografic ponderată (Fotheringham et al., 2002)

```
>library(spgwr)
>rsp<-as(rs,"SpatialPixelsDataFrame")
>gwr.band <- gwr.sel(formula(lm1), data=temp.co)
>gwr.fit <- gwr(formula(lm1), temp.co,
bandwidth=gwr.band)
>gwr.r <- gwr(formula(lm1), temp.co,
bandwidth=gwr.band,
fit.points = rsp, predict=TRUE)
rgwr<-raster(gwr.r$SDF['GWR'])

>plot(rgwr,col=rev(heat.colors(14)))
>plot(temp.co,add=T,pch=19,cex=0.5)
>text(coordinates(temp.co)[,1],
coordinates(temp.co)[,2],
temp.co$temp_med1,cex=0.7,pos=4)
```



► Modelul aditiv generalizat (Wood and Augustin, 2002)

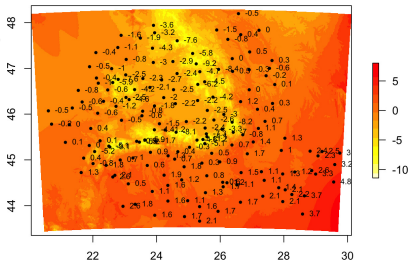
```
>library(mgcv)
>s.gam<-gam(temp_med1~s(dem)+s(lon)+s(lat)+
s(TWI), data=temp.co, select=T,method="REML")
>summary(s.gam)$r.sq
[1] 0.928
>rgam<-predict(rs,s.gam)
```

Analiza de regresie

► Regresia geografic ponderată (Fotheringham et al., 2002)

```
>library(spgwr)
>rsp<-as(rs,"SpatialPixelsDataFrame")
>gwr.band <- gwr.sel(formula(lm1), data=temp.co)
>gwr.fit <- gwr(formula(lm1), temp.co,
bandwidth=gwr.band)
>gwr.r <- gwr(formula(lm1), temp.co,
bandwidth=gwr.band,
fit.points = rsp, predict=TRUE)
rgwr<-raster(gwr.r$SDF['GWR'])

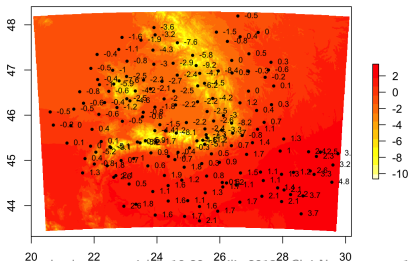
>plot(rgwr,col=rev(heat.colors(14)))
>plot(temp.co,add=T,pch=19,cex=0.5)
>text(coordinates(temp.co)[,1],
coordinates(temp.co)[,2],
temp.co$temp_med1,cex=0.7,pos=4)
```



► Modelul aditiv generalizat (Wood and Augustin, 2002)

```
>library(mgcv)
>g.gam<-gam(temp_med1~s(dem)+s(lon)+s(lat)+
s(TWI), data=temp.co, select=T,method="REML")
>summary(g.gam)$r.sq
[1] 0.928
>rgam<-predict(rs,g.gam)

>plot(rgam,col=rev(heat.colors(14)))
>plot(temp.co,add=T,pch=19,cex=0.5)
>text(coordinates(temp.co)[,1],
coordinates(temp.co)[,2],
temp.co$temp_med1,cex=0.7,pos=4)
```



Metode de interpolare

Biblioteci: **gstat**, **fields**, **automap**, **intamap**, **geoR**, **akima**,....

Aplicații ale limbajului
R în procesarea datelor
geospațiale

Alexandru Dumitrescu

Introducere

I/O datelor geospațiale

Proiecții și transformări
în sisteme de
coordonate

Vizualizare

Prelucrare

Analiza de regresie

Metode de interpolare

Produse

Concluzii

Bibliografie

Metode de interpolare

Biblioteci: **gstat**, **fields**, **automap**, **intamap**, **geoR**, **akima**,....

- ✓ Aproximativ toate metodele de interpolare care sunt folosite în mod curent sunt implementate în R prin bibliotecile suplimentare

Aplicații ale limbajului
R în procesarea datelor
geospațiale

Alexandru Dumitrescu

Introducere

I/O datelor geospațiale

Proiecții și transformări
în sisteme de
coordonate

Vizualizare

Prelucrare

Analiza de regresie

Metode de interpolare

Produse

Concluzii

Bibliografie

Metode de interpolare

Biblioteci: **gstat**, **fields**, **automap**, **intamap**, **geoR**, **akima**,....

- ✓ Aproximativ toate metodele de interpolare care sunt folosite în mod curent sunt implementate în R prin bibliotecile suplimentare
- ✓ Anumite metode sunt disponibile doar în R: **akima**, **3Dkriging** (metodă care ține cont în interpolare și de componenta timp, în cazul datelor cu dimensiune spațio-temporală)

Aplicații ale limbajului
R în procesarea datelor
geospațiale

Alexandru Dumitrescu

Introducere

I/O datelor geospațiale

Proiecții și transformări
în sisteme de
coordonate

Vizualizare

Prelucrare

Analiza de regresie

Metode de interpolare

Produse

Concluzii

Bibliografie

Metode de interpolare

Biblioteci: **gstat**, **fields**, **automap**, **intamap**, **geoR**, **akima**,....

- ✓ Aproximativ toate metodele de interpolare care sunt folosite în mod curent sunt implementate în R prin bibliotecile suplimentare
- ✓ Anumite metode sunt disponibile doar în R: **akima**, **3Dkriging** (metodă care ține cont în interpolare și de componenta timp, în cazul datelor cu dimensiune spațio-temporală)
- ✓ În cazul metodelor din familia Kriging, sunt implementate rutine de construirea automată a semi-variogramelor: funcția *autofitVariogram* din biblioteca **automap**

Metode de interpolare

Biblioteci: **gstat**, **fields**, **automap**, **intamap**, **geoR**, **akima**,....

- ✓ Aproximativ toate metodele de interpolare care sunt folosite în mod curent sunt implementate în R prin bibliotecile suplimentare
- ✓ Anumite metode sunt disponibile doar în R: **akima**, **3Dkriging** (metodă care ține cont în interpolare și de componenta timp, în cazul datelor cu dimensiune spațio-temporală)
- ✓ În cazul metodelor din familia Kriging, sunt implementate rutine de construirea automată a semi-variogramelor: funcția *autofitVariogram* din biblioteca **automap**
- ▶ IDW - metoda de interpolare prin ponderare în funcție de inversul distanței (Johnston et al., 2001)

```
>library(gstat)
>ridw<-idw(temp_med1~1, temp.co,rs, idp=3)
```

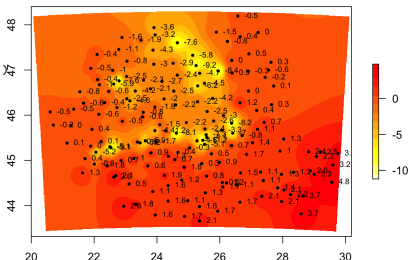
Metode de interpolare

Biblioteci: **gstat**, **fields**, **automap**, **intamap**, **geoR**, **akima**,....

- ✓ Aproximativ toate metodele de interpolare care sunt folosite în mod curent sunt implementate în R prin bibliotecile suplimentare
- ✓ Anumite metode sunt disponibile doar în R: **akima**, **3Dkriging** (metodă care ține cont în interpolare și de componenta timp, în cazul datelor cu dimensiune spațio-temporală)
- ✓ În cazul metodelor din familia Kriging, sunt implementate rutine de construirea automată a semi-variogramelor: funcția *autofitVariogram* din biblioteca **automap**
- IDW - metoda de interpolare prin ponderare în funcție de inversul distanței (Johnston et al., 2001)

```
>library(gstat)
>ridw<-idw(temp_med1-1, temp.co,rsp,idp=3)
```

```
>plot(raster(ridw)[[1]],col=rev(heat.colors(14)))
>plot(temp.co,add=T,pch=19,cex=0.5)
>text(coordinates(temp.co)[,1],
coordinates(temp.co)[,2],
temp.co$temp_med1,cex=0.7,pos=4)
```



Metode de interpolare

- ▶ TPS - thin plate spline (De Smith et al., 2007)

Metode de interpolare

► TPS - thin plate spline (De Smith et al., 2007)

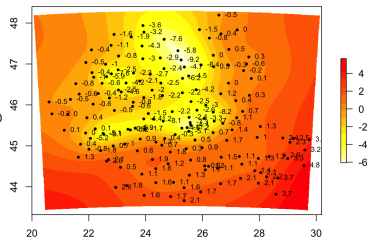
```
>library(fields)
>tps<-Tps(coordinates(temp.co),
temp.co$temp_medl)
>tps.v<-predict(tps,coordinates(rsp))
>rsp@data[, 'TPS']<-tps.v
```

Metode de interpolare

► TPS - thin plate spline (De Smith et al., 2007)

```
>library(fields)
>tps<-Tps(coordinates(temp.co),
temp.co$temp_medl)
>tps.v<-predict(tps,coordinates(rsp))
>rsp@data[, 'TPS']<-tps.v

>plot(raster(rsp['TPS']),col=rev(heat.colors(14)))
>plot(temp.co,add=T,pch=19,cex=0.5)
>text(coordinates(temp.co)[,1],
coordinates(temp.co)[,2],
temp.co$temp_medl,cex=0.7,pos=4)
```

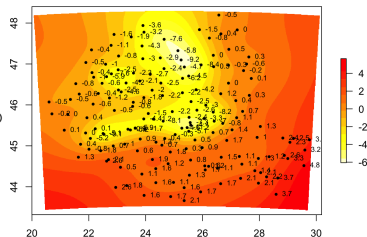


Metode de interpolare

► TPS - thin plate spline (De Smith et al., 2007)

```
>library(fields)
>tps<-Tps(coordinates(temp.co),
temp.co$temp_medl)
>tps.v<-predict(tps,coordinates(rsp))
>rsp@data[, 'TPS']<-tps.v

>plot(raster(rsp['TPS']),col=rev(heat.colors(14)))
>plot(temp.co,add=T,pch=19,cex=0.5)
>text(coordinates(temp.co)[,1],
coordinates(temp.co)[,2],
temp.co$temp_medl,cex=0.7,pos=4)
```



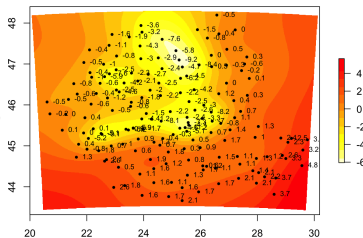
► Kriging (Pebesma, 2004)

Metode de interpolare

► TPS - thin plate spline (De Smith et al., 2007)

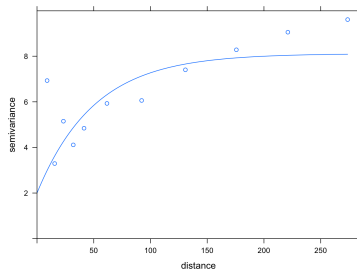
```
>library(fields)
>tps<-Tps(coordinates(temp.co),
temp.co$temp_med1)
>tps.v<-predict(tps,coordinates(rsp))
>rsp@data[, 'TPS']<-tps.v

>plot(raster(rsp['TPS']),col=rev(heat.colors(14)))
>plot(temp.co,add=T,pch=19,cex=0.5)
>text(coordinates(temp.co)[,1],
coordinates(temp.co)[,2],
temp.co$temp_med1,cex=0.7,pos=4)
```



► Kriging (Pebesma, 2004)

```
>library(automap)
>v<-autofitVariogram(temp_med1~1, temp.co,
model=c("Exp"))
>plot(v)
```

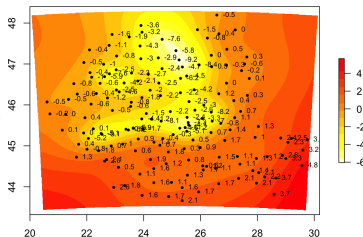


Metode de interpolare

► TPS - thin plate spline (De Smith et al., 2007)

```
>library(fields)
>tps<-Tps(coordinates(temp.co),
temp.co$temp_med1)
>tps.v<-predict(tps,coordinates(rsp))
>rsp@data[, 'TPS']<-tps.v

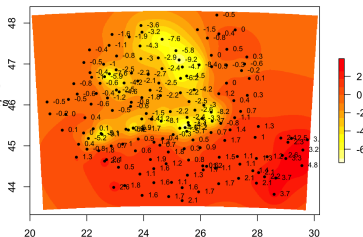
>plot(raster(rsp['TPS']), col=rev(heat.colors(14)))
>plot(temp.co,add=T,pch=19,cex=0.5)
>text(coordinates(temp.co)[,1],
coordinates(temp.co)[,2],
temp.co$temp_med1,cex=0.7,pos=4)
```



► Kriging (Pebesma, 2004)

```
>library(automap)
>v<-autofitVariogram(temp_med1-1, temp.co,
model=c("Exp"))
>plot(v)

>rkrig<-krige(temp_med1-1, temp.co,rsp,v$var_model)
>plot(raster(rkrig['var1.pred']),col=rev(heat.colors(14)))
```

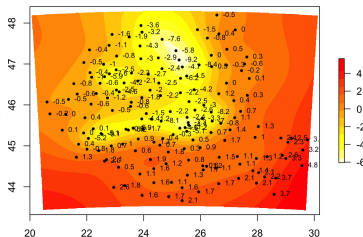


Metode de interpolare

► TPS - thin plate spline (De Smith et al., 2007)

```
>library(fields)
>tps<-Tps(coordinates(temp.co),
temp.co$temp_medl)
>tps.v<-predict(tps,coordinates(rsp))
>rsp@data[, 'TPS']<-tps.v

>plot(raster(rsp['TPS']), col=rev(heat.colors(14)))
>plot(temp.co,add=T,pch=19,cex=0.5)
>text(coordinates(temp.co)[,1],
coordinates(temp.co)[,2],
temp.co$temp_medl,cex=0.7,pos=4)
```

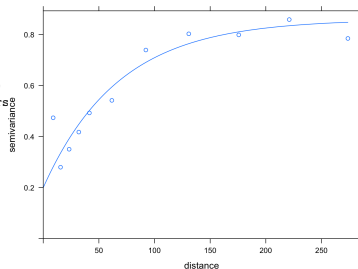


► Kriging (Pebesma, 2004)

```
>library(automap)
>v<-autofitVariogram(temp_medl-1, temp.co,
model=c("Exp"))
>plot(v)

>rkrig<-krige(temp_medl-1, temp.co,rsp,v$var_model)
>plot(raster(rkrig['vari.pred']),col=rev(heat.colors(14)))

>v<-autofitVariogram(temp_medl-dem+lat+lon+TWI,
temp.co,model=c("Exp"))
>plot(v)
```

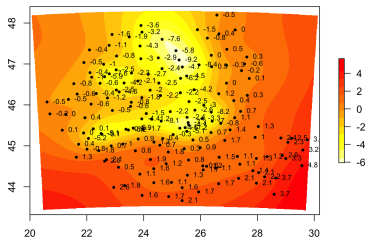


Metode de interpolare

► TPS - thin plate spline (De Smith et al., 2007)

```
>library(fields)
>tps<-Tps(coordinates(temp.co),
temp.co$temp_medl)
>tps.v<-predict(tps,coordinates(rsp))
>rsp@data[, 'TPS']<-tps.v

>plot(raster(rsp['TPS']), col=rev(heat.colors(14)))
>plot(temp.co,add=T,pch=19,cex=0.5)
>text(coordinates(temp.co)[,1],
coordinates(temp.co)[,2],
temp.co$temp_medl,cex=0.7,pos=4)
```



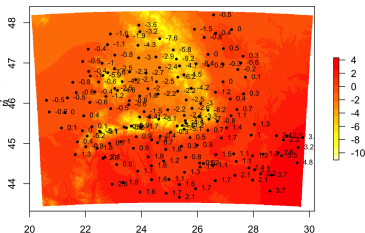
► Kriging (Pebesma, 2004)

```
>library(automap)
>v<-autofitVariogram(temp_medl-1, temp.co,
model=c("Exp"))
>plot(v)

>rkrig<-krige(temp_medl-1, temp.co,rsp,v$var_model)
>plot(raster(rkrig['var1.pred']),col=rev(heat.colors(14)))

>v<-autofitVariogram(temp_medl-dem+lat+lon+TWI,
temp.co,model=c("Exp"))
>plot(v)

>rskrig<-krige(temp_medl-dem+lat+lon+TWI,
temp.co,rsp,v$var_model)
>plot(raster(rskrig['var1.pred']),
col=rev(heat.colors(14)))
```



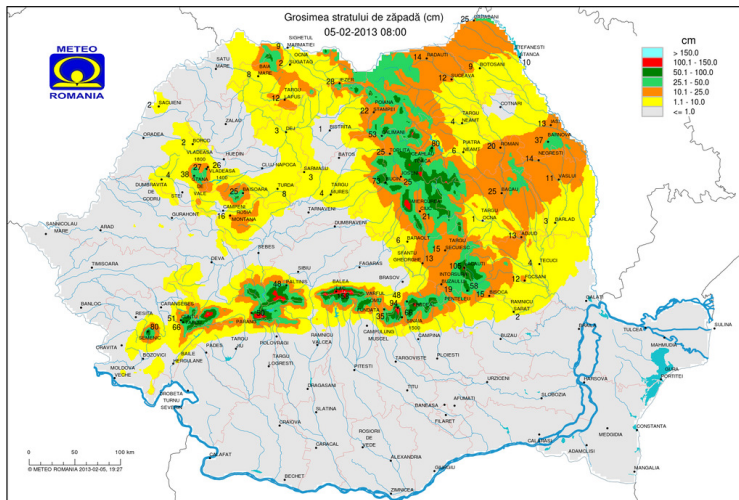


Figura: sursa:http://www.inmh.ro/images/clima/SZA_orar_interpolat.png

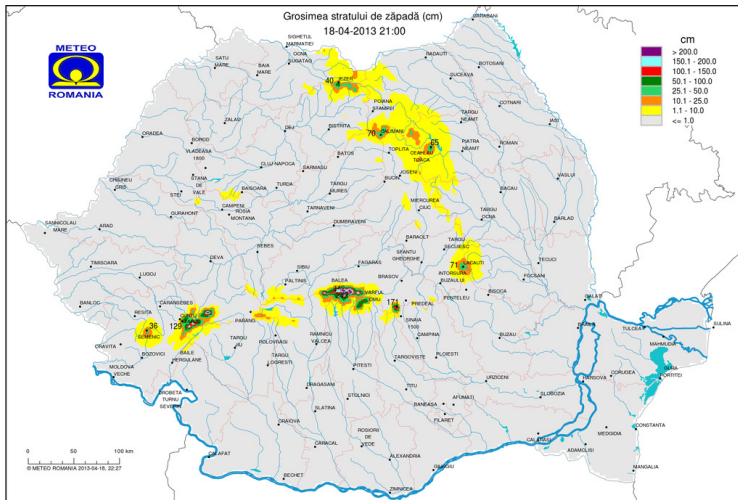


Figura: sursa: http://www.inmh.ro/images/clima/SZA_orar_interpolat.png

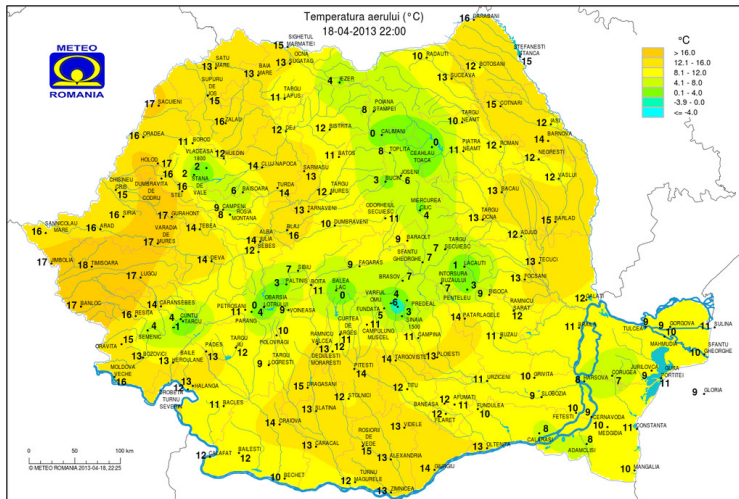


Figura: sursa: http://www.meteoromania.ro/images/clima/temperatura_orara.png

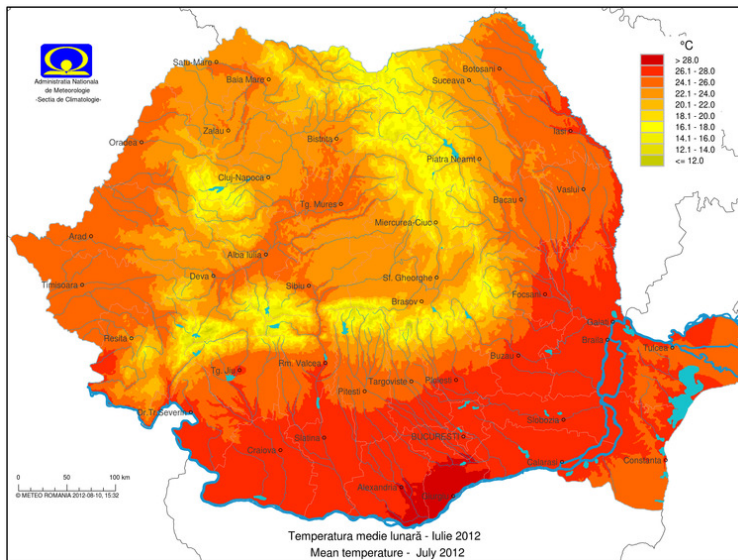


Figura: sursa: http://www.meteoromania.ro/anm/?page_id=2279

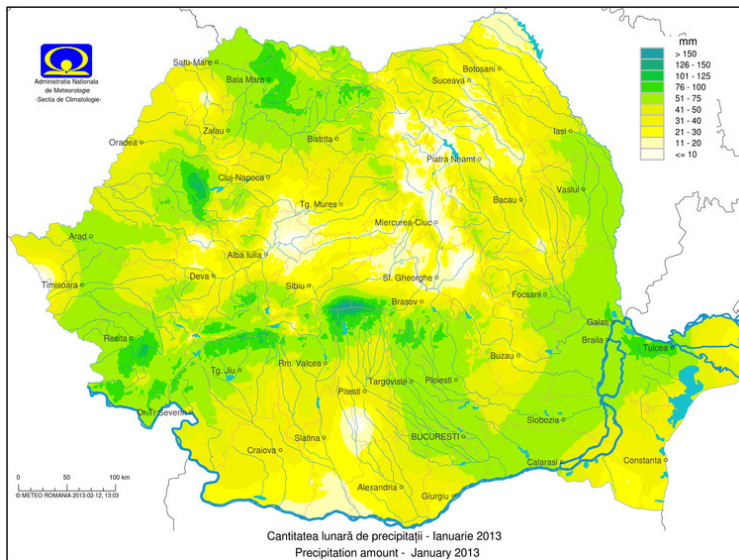


Figura: sursa: http://www.meteoromania.ro/anm/?page_id=2279

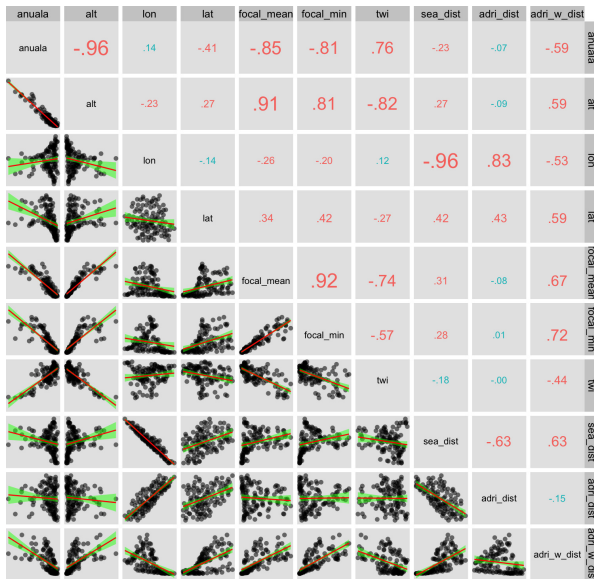


Figura: Matricea graficelor de corelație de tip puncte dintre temperatura medie anuală (1961-1990) și variabilele dependente. Coeficienții de corelație Pearson reprezentați prin culoare roșie indică un nivel de semnificație statistică $p < 0.05$

Concluzii

R plus:

Aplicații ale limbajului
R în procesarea datelor
geospațiale

Alexandru Dumitrescu

Introducere

I/O datelor geospațiale

Proiecții și transformări
în sisteme de
coordonate

Vizualizare

Prelucrare

Analiza de regresie

Metode de interpolare

Produse

Concluzii

Bibliografie

Concluzii

R plus:

- ✓ Automatizarea tuturor procedurilor specifice unui SIG (import date, procesare, realizare layout, export hartă finală ca imagine);

Aplicații ale limbajului
R în procesarea datelor
geospațiale

Alexandru Dumitrescu

Introducere

I/O datelor geospațiale

Proiecții și transformări
în sisteme de
coordonate

Vizualizare

Prelucrare

Analiza de regresie

Metode de interpolare

Produse

Concluzii

Bibliografie

Concluzii

R plus:

- ✓ Automatizarea tuturor procedurilor specifice unui SIG (import date, procesare, realizare layout, export hartă finală ca imagine);
- ✓ Open-source - codul este disponibil, oferă posibilitatea reutilizării/modificării funcțiilor deja existente;

Concluzii

R plus:

- ✓ Automatizarea tuturor procedurilor specifice unui SIG (import date, procesare, realizare layout, export hartă finală ca imagine);
- ✓ Open-source - codul este disponibil, oferă posibilitatea reutilizării/modificării funcțiilor deja existente;
- ✓ Comunitate de utilizatori foarte mare, în continuă creștere (<http://rseek.org>, <https://stat.ethz.ch/mailman/listinfo/r-sig-geo>);

Concluzii

R plus:

- ✓ Automatizarea tuturor procedurilor specifice unui SIG (import date, procesare, realizare layout, export hartă finală ca imagine);
- ✓ Open-source - codul este disponibil, oferă posibilitatea reutilizării/modificării funcțiilor deja existente;
- ✓ Comunitate de utilizatori foarte mare, în continuă creștere (<http://rseek.org>, <https://stat.ethz.ch/mailman/listinfo/r-sig-geo>);
- ✓ Portabilitate - se poate utiliza pe orice sistem de operare existent.

Aplicații ale limbajului
R în procesarea datelor
geospațiale

Alexandru Dumitrescu

Introducere

I/O datelor geospațiale

Proiecții și transformări
în sisteme de
coordonate

Vizualizare

Prelucrare

Analiza de regresie

Metode de interpolare

Produse

Concluzii

Bibliografie

Concluzii

R plus:

- ✓ Automatizarea tuturor procedurilor specifice unui SIG (import date, procesare, realizare layout, export hartă finală ca imagine);
- ✓ Open-source - codul este disponibil, oferă posibilitatea reutilizării/modificării funcțiilor deja existente;
- ✓ Comunitate de utilizatori foarte mare, în continuă creștere (<http://rseek.org>, <https://stat.ethz.ch/mailman/listinfo/r-sig-geo>);
- ✓ Portabilitate - se poate utiliza pe orice sistem de operare existent.

R minus:

- ✓ Utilizare interactivă: zoom, pan, editarea datelor vector;

Concluzii

R plus:

- ✓ Automatizarea tuturor procedurilor specifice unui SIG (import date, procesare, realizare layout, export hartă finală ca imagine);
- ✓ Open-source - codul este disponibil, oferă posibilitatea reutilizării/modificării funcțiilor deja existente;
- ✓ Comunitate de utilizatori foarte mare, în continuă creștere (<http://rseek.org>, <https://stat.ethz.ch/mailman/listinfo/r-sig-geo>);
- ✓ Portabilitate - se poate utiliza pe orice sistem de operare existent.

R minus:

- ✓ Utilizare interactivă: zoom, pan, editarea datelor vector;
- ✓ Exista mai multor clase și obiecte dedicate aceluși tipuri de date (ex: "RasterBrick" vs. "SpatialPixelsDataFrame");

Concluzii

R plus:

- ✓ Automatizarea tuturor procedurilor specifice unui SIG (import date, procesare, realizare layout, export hartă finală ca imagine);
- ✓ Open-source - codul este disponibil, oferă posibilitatea reutilizării/modificării funcțiilor deja existente;
- ✓ Comunitate de utilizatori foarte mare, în continuă creștere (<http://rseek.org>, <https://stat.ethz.ch/mailman/listinfo/r-sig-geo>);
- ✓ Portabilitate - se poate utiliza pe orice sistem de operare existent.

R minus:

- ✓ Utilizare interactivă: zoom, pan, editarea datelor vector;
- ✓ Exista mai multor clase și obiecte dedicate aceluși tipuri de date (ex: "RasterBrick" vs. "SpatialPixelsDataFrame");
- ✓ Funcționalitatea maximă poate fi atinsă strict prin utilizarea R în linie de comandă.

Concluzii

R plus:

- ✓ Automatizarea tuturor procedurilor specifice unui SIG (import date, procesare, realizare layout, export hartă finală ca imagine);
- ✓ Open-source - codul este disponibil, oferă posibilitatea reutilizării/modificării funcțiilor deja existente;
- ✓ Comunitate de utilizatori foarte mare, în continuă creștere (<http://rseek.org>, <https://stat.ethz.ch/mailman/listinfo/r-sig-geo>);
- ✓ Portabilitate - se poate utiliza pe orice sistem de operare existent.

R minus:

- ✓ Utilizare interactivă: zoom, pan, editarea datelor vector;
- ✓ Exista mai multor clase și obiecte dedicate aceluși tipuri de date (ex: "RasterBrick" vs. "SpatialPixelsDataFrame");
- ✓ Funcționalitatea maximă poate fi atinsă strict prin utilizarea R în linie de comandă.

Simon Blomberg: This is R. There is no if. Only how.

Evelyn Hall and Simon 'Yoda' Blomberg
R-help (April 2005)

Mulțumesc
pentru
atenție

Aplicații ale limbajului
R în procesarea datelor
geospațiale

Alexandru Dumitrescu

Introducere

I/O datelor geospațiale

Proiecții și transformări
în sisteme de
coordonate

Vizualizare

Prelucrare

Analiza de regresie

Metode de interpolare

Produce

Concluzii

Bibliografie

- De Smith, M., M. Goodchild, and P. Longley, 2007: *Geospatial analysis: a comprehensive guide to principles, techniques and software tools*. Troubador Publishing.
- Fotheringham, A., C. Brunsdon, and M. Charlton, 2002: *Geographically weighted regression: the analysis of spatially varying relationships*. John Wiley & Sons Inc.
- Johnston, K., J. Ver Hoef, K. Krivoruchko, and N. Lucas, 2001: *Using ArcGIS Geostatistical Analyst*, Vol. 300. Esri New York.
- Pebesma, E. J., 2004: Multivariable geostatistics in S the gstat package. *Computers & Geosciences*, **30**, 683–691.
- Thom, H. C. S., 1966: Some methods of climatological analysis. WMO Technical Note 81, World Meteorological Organisation, 53 pp. [WMO - No.199. TP. 103].
- Venables, W. and B. Ripley, 1999: *Modern Applied Statistics with S-Plus*. Springer.
- Wood, S. N. and N. H. Augustin, 2002: GAMs with integrated model selection using penalized regression splines and applications to environmental modelling. *Ecological Modelling*, **157** (2-3), 157 – 177, doi:DOI:10.1016/S0304-3800(02)00193-X.