

Анализ данных с использованием GRASS GIS и R

[Обсудить в форуме](#) Комментариев — 12

Эта страница опубликована в основном списке статей сайта по адресу <http://gis-lab.info/qa/grass-r.html>

Примеры совместного использования ГИС GRASS и статпакета и языка R

ГИС GRASS является мощной геоинформационной системой с открытым исходным кодом, предназначенной для управления пространственными данными, обработки изображений (в том числе данных ДЗЗ), пространственного моделирования, визуализации данных и т.д. ГИС GRASS предоставляет пользователю множество модулей и функций, облегчающих анализ и обработку данных. R представляет собой платформу с открытым исходным кодом, предназначенную для статистических вычислений, моделирования и анализа. В данной статье рассматриваются примеры совместного использования R и ГИС GRASS.

Цель этой статьи - дать читателю отправную точку для дальнейшего изучения и использования связки GRASS GIS и R. Предполагается, что читатель имеет некоторое знакомство с данными системами. В частности, читатель должен иметь представление о синтаксисе языка R и иметь хотя бы небольшой опыт работы с ГИС GRASS.

Хотя для экспериментов мы будем использовать [набор данных geosample](#), жесткой привязки к данному набору данных нет, и читатель с легкостью может адаптировать примеры под свои нужды и данные.

В статье использованы материалы книги ["Open Source GIS: A GRASS GIS Approach"](#) (авторы Markus Neteler, Helena Mitasova).

Содержание

- [1 Установка необходимых пакетов](#)
- [2 Чтение и запись данных GRASS в среде R](#)
 - [2.1 Первые шаги](#)
 - [2.2 Импорт векторных данных](#)
 - [2.3 Импорт растровых данных](#)
 - [2.4 Экспорт данных в GRASS GIS](#)
- [3 Пример обработки данных](#)

Установка необходимых пакетов

Предположим, что у нас есть база геоданных GRASS (location/mapset), которую мы хотим прочитать в R, определенным образом обработать/проанализировать и сохранить полученный результат обратно в базу геоданных. Для решения этой задачи существует несколько подходов, мы воспользуемся тем, что и GRASS GIS, и R предоставляют пользователю командную оболочку, в которой пользователь может вводить команды и немедленно получать результат. Таким образом, режим работы будет следующий:

- Заходим в нужную область (location/mapset) GRASS GIS;
- Из командной строки GRASS вызываем среду R;
- Производим требуемые манипуляции с геоданными GRASS, используя возможности анализа среды R;
- При необходимости сохраняем результаты;
- Выходим из R и GRASS.

Таким образом, большую часть времени мы будем находиться одновременно в двух системах: GRASS и R. Однако, чтобы такого рода "матрёшка" сработала, необходимо, чтобы в среде R были установлены пакеты, позволяющие читать данные, хранящиеся в формате GRASS.

Для работы нам понадобится пакет `spgrass6` и его зависимости (однако для того, чтобы производить анализ пространственных данных, рекомендуется посмотреть также пакеты `akima`, `fields`, `geoR`, `grid`, `gstat`, `lattice`, `MASS`, `scatterplot3d`, `spatial` и `stepfun`).

Установка пакетов в среде R не должна вызывать каких-либо трудностей, например для установки пакетов `spgrass6` и `gstat` достаточно выполнить следующие команды (требуется подключение к интернет):

```
install.packages("spgrass6", "gstat", dependencies = TRUE)
```

Загрузить необходимую библиотеку (например, `spgrass6`) в среду R можно командой

```
library(spgrass6)
```

Чтение и запись данных GRASS в среде R

Первые шаги

Запустим GRASS и из командной строки GRASS загружаем R. В конечном итоге перед нами окажется командная строка системы R.

Загрузим в R необходимые библиотеки:

```
library(spgrass6)
```

После загрузки данной библиотеки уже можно взаимодействовать с GRASS GIS. Например, введем следующую команду:

```
str(gmeta6())
```

Результат выполнения команды приводится ниже:

```
> str(gmeta6())
List of 24
 $ GISDBASE      : chr "/home/dima/GIS/grass"
 $ LOCATION_NAME: chr "grass"
 $ MAPSET        : chr "PERMANENT"
 $ MONITOR        : chr "x0"
 $ GRASS_GUI      : chr "text"
 $ n              : num 57.3
 $ s              : num 49
 $ w              : num 75
 $ e              : num 90
 $ t              : num 1
 $ b              : num 0
 $ nsres          : num 0.00417
 $ nsres3         : num 1.04
 $ ewres          : num 0.00417
 $ ewres3         : num 1
 $ tbres          : num 1
 $ rows           : int 1992
 $ rows3          : int 8
 $ cols           : int 3600
 $ cols3          : int 15
 $ depths         : int 1
 $ cells          : chr "7171200"
```

```
$ 3dcells      : chr "120"
$ proj4        : chr "+proj=longlat +a=6378137 +rf=298.257223563 +no_defs
+towgs84=0.000,0.000,0.000"
- attr(*, "class")= chr "gmeta6"
```

Как легко заменить, данная команда вывела информацию о текущей области (region) GRASS и другие параметры.

Находясь в R, вы имеете возможность запускать команды GRASS, не выходя из сессии R. Для этой цели можно использовать команду `system`, например:

```
> system("g.region -p")
projection: 3 (Latitude-Longitude)
zone:      0
datum:     wgs84
ellipsoid: wgs84
north:     57:18N
south:     49N
west:      75E
east:      90E
nsres:     0:00:15
ewres:     0:00:15
rows:      1992
cols:      3600
cells:     7171200
```

Импорт векторных данных

Импортируем несколько карт из GRASS GIS в среду R, для этого используется команда `readVECT6`.

```
admin <- readVECT6("admin", ignore.stderr=TRUE)
```

Посмотрим информацию:

```
> summary(admin)
Object of class SpatialPolygonsDataFrame
Coordinates:
      min      max
x 75.08194 89.86998
y 49.08308 57.25027
Is projected: FALSE
proj4string :
+proj=longlat +ellps=WGS84 +datum=WGS84 +no_defs +towgs84=0,0,0
Data attributes:
      cat      NAME
Min.   :1.00   Алтайский край      :1
1st Qu.:1.75   Новосибирская область:1
Median :2.50   Республика Алтай     :2
Mean    :2.50
3rd Qu.:3.25
Max.    :4.00
```

Аналогично, загрузим данные по почвам:

```
> soils <- readVECT6("soils", ignore.stderr=TRUE)
> summary(soils)
```

Object of class SpatialPolygonsDataFrame

Coordinates:

min max

x 75.08194 89.86998

y 49.08308 57.25027

Is projected: FALSE

proj4string :

+proj=longlat +ellps=WGS84 +datum=WGS84 +no_defs +towgs84=0,0,0

Data attributes:

	cat	FAOSOIL	UNIT	TYPE	NAME
Min. :	1	Ch1-3a : 16	I :39	LITHOSOLS :39	Черноземы:52
1st Qu.:	44	C1-3a : 14	Ch :17	Haplic Chernozems:17	Литосоли :39
Median :	87	Sm13-3a: 14	C :16	CHERNOZEMS :16	Солонцы :18
Mean :	87	Oe1-a : 10	Sm :15	Mollic Solonetz :15	Сероземы :11
3rd Qu.:	130	Mo1-2ab: 9	Ck :10	Calcic Chernozems:10	Гистосоли:10
Max. :	173	(Other):110	(Other):76	(Other) :76	(Other) :43
NA's :	1	NA's : 1	NA's : 1	NA's : 1	NA's : 1

FAOSOIL_L

Min. : 1.00

1st Qu.: 8.00

Median :21.00

Mean :20.26

3rd Qu.:34.00

Max. :40.00

NA's : 1.00

При необходимости можно отобразить данные на карте:

```
plot(admin, axes=TRUE)
```

Добавим еще один слой:

```
plot(soils, add=TRUE, col="red")
```

Импорт растровых данных

Импорт растровых данных производится аналогично. Для этого предназначена команда readRAST6:

```
readRAST6(vname, cat=NULL, ignore.stderr = FALSE, NODATA=NULL, plugin=NULL,
mapset=NULL, useGDAL=TRUE, close_OK=TRUE)
```

В первую очередь, нас будет интересовать параметр vname - название растровой карты GRASS. Считаем, к примеру, растр под названием modis:

```
> modis <- readRAST6("modis", ignore.stderr=TRUE)
> summary(modis)
Object of class SpatialGridDataFrame
Coordinates:
  min max
x 75 90.0
y 49 57.3
Is projected: FALSE
proj4string :
+proj=longlat +a=6378137 +rf=298.257223563 +no_defs
+towgs84=0.000,0.000,0.000
Number of points: 2
Grid attributes:
  cellcentre.offset  cellsize  cells.dim
x          75.00208  0.004166667      3600
y          49.00208  0.004166667      1992
Data attributes:
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	NA's
0	4328	5418	7028	7499	32770	209650

Экспорт данных в GRASS GIS

Для экспорта растровых данных существует команда:

```
writeRAST6(x, vname, zcol = 1, NODATA=NULL, ignore.stderr = FALSE, useGDAL=TRUE,
overwrite=FALSE, flags=NULL)
```

Здесь x - фрейм пространственных данных, который будет сохранен как растровая карта GRASS под именем vname (подробнее см. в справке R).

Аналогично, для работы с векторными данными используется команда:

```
writeVECT6(SDF, vname, v.in.ogr_flags=NULL, ignore.stderr = FALSE)
```

Здесь SDF - фрейм пространственных данных, который будет сохранен как векторная карта GRASS под именем vname (подробнее см. в справке R).

Пример обработки данных

В наборе geosample содержатся несколько растровых изображений: это данные MODIS (растр modis) и данные рельефа (растр relief). В качестве примера построим регрессию зависимости яркости modis от высоты relief.

Сначала загрузим разрешение, чтобы не обрабатывать излишнее количество данных, сделаем это в GRASS (хотя это же можно и через R):

```
> g.region res=0:01:0 -p
projection: 3 (Latitude-Longitude)
zone:      0
datum:     wgs84
ellipsoid: wgs84
north:     57:18N
south:     49N
west:      75E
east:      90E
nsres:     0:01
ewres:     0:01
rows:      498
cols:      900
cells:     448200
```

Заходим в R, подключаем библиотеки, импортируем данные:

```
> library(spgrass6)
> modis <- readRAST6("modis", ignore.stderr=TRUE)
> relief <- readRAST6("relief", ignore.stderr=TRUE)
```

Приведем данные к числовому виду, понятному функции для линейных регрессий lm:

```
> relief_mat <- as.matrix(relief)
```

```
> relief_vec <- as.vector(relief_mat)
> modis_mat <- as.matrix(modis)
> modis_vec <- as.vector(modis_mat)
```

Построим линейную регрессию:

```
> model <- lm(modis_vec ~ relief_vec)
> summary(model)
```

Call:

```
lm(formula = modis_vec ~ relief_vec)
```

Residuals:

Min	1Q	Median	3Q	Max
-15950.9	-2035.4	-182.4	1276.3	20414.5

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	4.625e+03	7.569e+00	611.1	<2e-16 ***
relief_vec	4.916e+00	9.682e-03	507.7	<2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3893 on 433741 degrees of freedom

(14457 observations deleted due to missingness)

Multiple R-squared: 0.3728, Adjusted R-squared: 0.3728

F-statistic: 2.578e+05 on 1 and 433741 DF, p-value: < 2.2e-16

Интересно сравнить результаты, которые мы получили в R с регрессией, которую мы построим в самой ГИС GRASS. Для этого вызовем команду GRASS r.regression.line:

```
> system("r.regression.line map1=modis map2=relief")
100%
y = a + b*x
a: offset
b: gain
R: sumXY - sumX*sumY/tot
N: number of elements
medX, medY: Means
sdX, sdY: Standard deviations
a b R N F medX sdX medY sdY
-44.4697 0.0758367 0.610568 433743 -0.372793 7025.41 4915.05 488.314 610.483
```

Построим теперь квадратичную регрессию, для вычисления которой в GRASS GIS нет соответствующего модуля:

```
> model2 <- lm(relief_vec ~ modis_vec + I(modis_vec^2))
> summary(model2)
```

Call:

```
lm(formula = relief_vec ~ modis_vec + I(modis_vec^2))
```

Residuals:

Min	1Q	Median	3Q	Max
-1930.25	-245.41	-157.72	88.94	2803.82

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	3.340e+02	2.308e+00	144.72	<2e-16	***
modis_vec	-1.310e-02	4.810e-04	-27.23	<2e-16	***
I(modis_vec^2)	3.351e-06	1.730e-08	193.70	<2e-16	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 463.8 on 433740 degrees of freedom

(14457 observations deleted due to missingness)

Multiple R-squared: 0.4227, Adjusted R-squared: 0.4227

F-statistic: 1.588e+05 on 2 and 433740 DF, p-value: < 2.2e-16

Таким образом, на нескольких простых примерах было показан основной цикл работы в двух средах GRASS GIS и R.

[Обсудить в форуме](#) Комментариев — 12

Последнее обновление: 2014-05-15 01:34

Дата создания: 06.07.2010

Автор(ы): [Дмитрий Колесов](#)