

A7 Die R-Pakete sp und spdep

Jan-Philipp Kolb

22 Oktober 2018

Themen dieses Abschnitts

- Eine räumliche Stichprobe ziehen
- Adressen für die gezogenen Punkte bestimmen
- Adressdatensatz bereinigen
- Wie lässt sich die Entfernung bestimmen

Das erste Gesetz der Geographie (TFLG)

“All things are related, but nearby things are more related than distant things” [Tobler, 1970]

Shapefile mit Regionalschlüssel herunterladen

```
library(rgdal)
```

```
setwd(vg250path)
```

```
VG250 <- readOGR ("VG250_GEM.shp", "VG250_GEM")
```

```
## OGR data source with driver: ESRI Shapefile
```

```
## Source: "D:\GESIS\data\vg250_3112.utm32s.shape.ebenen\vg250"
```

```
## with 11395 features
```

```
## It has 23 fields
```

Räumliche Stichprobe

- Mit der Funktion `spsample` aus dem Paket `sp` kann man eine räumliche Stichprobe ziehen.

```
spatsamp <- spsample(VG250, 100, type="random")
```

<code>type</code>	character; "random" for completely spatial random; "regular" for regular (systematically aligned) sampling; "stratified" for stratified random (one single random location in each "cell"); "nonaligned" for nonaligned systematic sampling (nx random y coordinates, ny random x coordinates); "hexagonal" for sampling on a hexagonal lattice; "clustered" for clustered sampling; "Fibonacci" for Fibonacci sampling on the sphere (see references).
-------------------	---

Point in Polygon

- Mit der Funktion `over` kann man feststellen in welchem Polygon ein Punkt liegt.

```
tmp <- sp::over(spatsamp, VG250)
```

```
head(tmp[,1:6])
```

##	ADE	GF	BSG	RS	AGS	SDV_RS
## 1	6	4	1	1004201111111	10042111	1004201111111
## 2	6	4	1	066340007007	06634007	066340007007
## 3	6	4	1	091755114128	09175128	091755114128
## 4	6	4	1	096760125125	09676125	096760125125
## 5	6	4	1	147130000000	14713000	147130000000
## 6	6	4	1	110000000000	11000000	110000000000

Koordinatenreferenzsystem (CRS)

CRS bei Wikipedia

- Ein Koordinatenreferenzsystem ist ein Koordinatensystem, das durch Verknüpfung mit einem Datum auf die reale Welt bezogen ist.
- EPSG-Codes sind ein weit verbreitetes Mittel, um den räumlichen Bezug von Geodaten eindeutig anzugeben.
- Die 4- bis 5-stelligen Schlüsselnummern beinhalten jeweils eine komplette Beschreibung eines Koordinatenreferenzsystems.
- Es gibt mehrere hundert solcher Referenzsysteme, und jedes besteht aus einer Liste von 10 bis 20 Parametern.

Daten in ein anderes CRS übertragen

spTransform for map projection and datum transformation

```
newData<-sp::spTransform(spatsamp, CRS("+init=epsg:3857"))
```

```
head(spatsamp)
```

```
## class      : SpatialPoints
## features   : 1
## extent     : 334718.6, 334718.6, 5475591, 5475591 (xmin,
## coord. ref. : +proj=utm +zone=32 +ellps=GRS80 +units=m +no
```

```
head(newData)
```

```
## class      : SpatialPoints
## features   : 1
## extent     : 748223.4, 748223.4, 6344811, 6344811 (xmin,
## coord. ref. : +init=epsg:3857 +proj=merc +a=6378137 +b=6378
```

Eine Karte von Afrika

```
library(maptools)
data(wrld_simpl)
Africa <- wrld_simpl[wrld_simpl@data$REGION==2,]
plot(Africa)
```



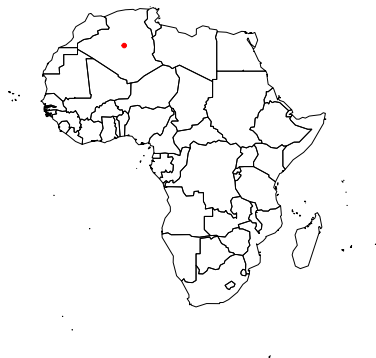
Das Zentrum eines Polygonzuges

```
Af <- coordinates(Africa)
head(Af)
```

```
##           [,1]           [,2]
## DZA  2.627813  28.1721102
## AGO 17.552463 -12.3503789
## BEN  2.332296   9.6047655
## COG 15.218362  -0.8732659
## COD 23.646032  -2.8711605
## BDI 29.901786  -3.3461606
```

Die Koordinaten plotten

```
plot(Africa)  
points(x=Af[1,1],y=Af[1,2],col="red",pch=20)
```



Die nächsten Nachbarn finden

```
library(spdep)
Af_nb <- tri2nb(Af)
```

Die Nachbarn für das erste Land (Algerien):

```
Af_nb[1]
```

```
## [[1]]
```

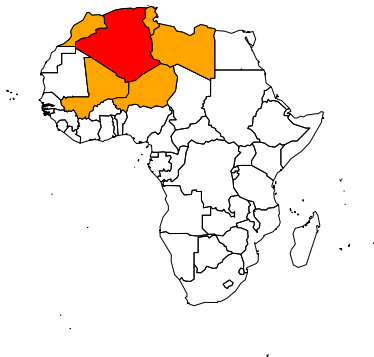
```
## [1] 24 26 27 32 48
```

```
Africa@data[Af_nb[[1]],1:5]
```

##	FIPS	ISO2	ISO3	UN	NAME
##	LBY	LY	LYB	434	Libyan Arab Jamahiriya
##	MLI	ML	MLI	466	Mali
##	MAR	MO	MAR	504	Morocco
##	NER	NG	NER	562	Niger
##	TUN	TS	TUN	788	Tunisia

Die Nachbarn finden

```
plot(Africa)
plot(Africa[1,],col="red",add=T)
plot(Africa[Af_nb[1][[1]],],col="orange",add=T)
```



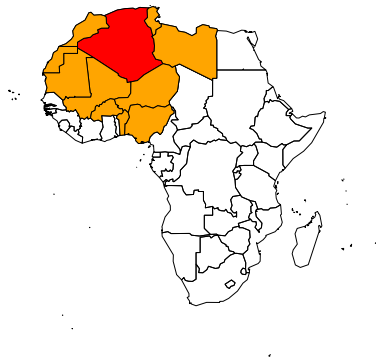
k nearest neighbours

```
IDs <- row.names(as(Africa, "data.frame"))  
(Af10_nb <- knn2nb(knearneigh(Af, k = 10), row.names = IDs))
```

```
## Neighbour list object:  
## Number of regions: 57  
## Number of nonzero links: 570  
## Percentage nonzero weights: 17.54386  
## Average number of links: 10  
## Non-symmetric neighbours list
```

Die 10 nächsten Nachbarn finden

```
plot(Africa)
plot(Africa[1,],col="red",add=T)
plot(Africa[Af10_nb[1][[1]],],col="orange",add=T)
```



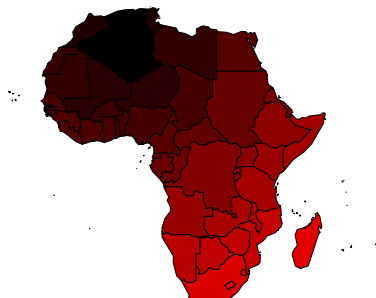
Die Distanz berechnen

```
Af <- coordinates(Africa) # get centroid  
library(raster)  
pointDistance(Af[1:4,], lonlat=TRUE) # compute distance
```

	[,1]	[,2]	[,3]	[,4]
## [1,]	0	NA	NA	NA
## [2,]	4763231	0	NA	NA
## [3,]	2055609	2954497	0	NA
## [4,]	3484053	1295173	1839191	0

Berechnen/zeichnen einer Distanzmatrix

```
Dist_Af <- pointDistance(Af, lonlat=TRUE)
Af_color <- Dist_Af[,1]
Af_color <- Af_color/max(Af_color)
Af_color <- rgb(Af_color,0,0)
plot(Africa,col=Af_color)
```



A7A Übung - Nachbarschaften in London

- Lade den Datensatz london_sport von meinem Github Verzeichnis herunter.
- Importiere den Datensatz.
- Bestimme die nächsten Nachbarn des Stadtteils *City of London*

```
setwd("D:/github/geocourse/data/")
london_sport <- readOGR("london_sport.shp", "london_sport")

## OGR data source with driver: ESRI Shapefile
## Source: "D:\github\geocourse\data\london_sport.shp", layer:
## with 33 features
## It has 4 fields
## Integer64 fields read as strings: Pop_2001
```

- **Raster, CMSAF and solaR**

<https://procomun.wordpress.com/2011/06/17/raster-cmsaf-and-solar/>

- **Getting rasters into shape from R**

<https://johnbaumgartner.wordpress.com/2012/07/26/getting-rasters-into-shape-from-r/>