

## A5 - Shapefiles

Jan-Philipp Kolb

23 Oktober 2018

# Worum geht es in diesem Abschnitt

- Shapefiles mit Vorwahl- und PLZ-Bereichen importieren
- PLZ-Bereiche zusammenfassen
- Eine räumliche Stichprobe ziehen
- Adressen für die gezogenen Punkte bestimmen
- Adressdatensatz bereinigen
- Entfernung zum Hauptbahnhof bestimmen

# Das shapefile Format ...

- ... ist ein beliebtes Format räumlicher Vektordaten für geographisches Informationssysteme (GIS).
- Es wurde entwickelt und reguliert von ESRI
- (meist) offene Spezifikation um Daten Interoperabilität zwischen Esri und anderen Formaten zu sichern.
- Es können Punkte, Linien und Polygone beschrieben werden
- Jedes Element hat Attribute, wie bspw. Name oder Temperatur die es beschreiben.

Quelle: <https://en.wikipedia.org/wiki/Shapefile>

# Der R Befehl `readShapePoly`

Um Shape-Dateien zu lesen, ist es notwendig, die drei Dateien mit den folgenden Dateierweiterungen im gleichen Verzeichnis zu haben:

- .shp
- .dbf
- .shx

# Vorwahlbereiche in Deutschland

<[www.bundesnetzagentur.de/](http://www.bundesnetzagentur.de/)>

- Wir verwenden das Paket `maptools`' um die Daten einzulesen:

```
setwd(geodata_path)
library(maptools)
onb <- readShapePoly("onb_grenzen.shp")
```

- Quelle Ortsnetzbereiche: **Bundesnetzagentur**

# Die Karte zeichnen

```
plot(onb)
```



# Der Datenslot

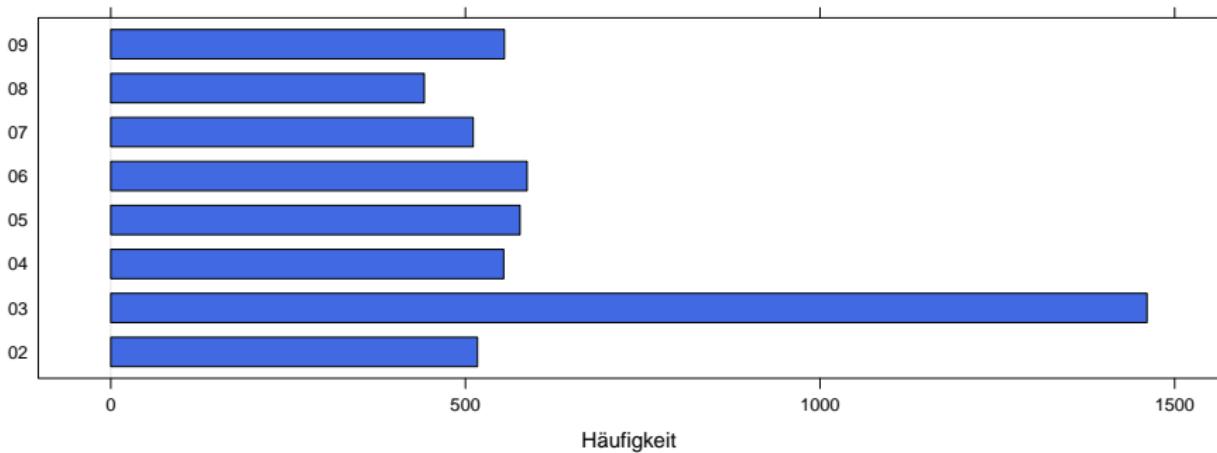
```
kable(head(onb@data))
```

|   | VORWAHL | NAME                 | KENNUNG |
|---|---------|----------------------|---------|
| 0 | 04651   | Sylt                 | NA      |
| 1 | 04668   | Klanxbüll            | NA      |
| 2 | 04664   | Neukirchen b Niebüll | NA      |
| 3 | 04663   | Süderlügum           | NA      |
| 4 | 04666   | Ladelund             | NA      |
| 5 | 04631   | Glücksburg Ostsee    | NA      |

# Einen Vorwahlbereich ausschneiden

```
vwb <- onb@data$VORWAHL  
vwb2 <- substr(vwb, 1,2)
```

```
library(lattice)  
barchart(table(vwb2), col="royalblue",  
         xlab="Häufigkeit")
```



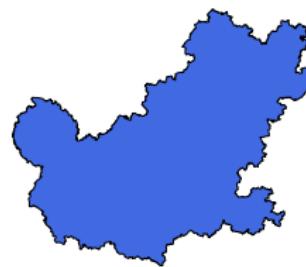
# Vorwahlbereich ausschneiden

```
vwb6 <- onb[vwb2=="06",]  
plot(vwb6)
```



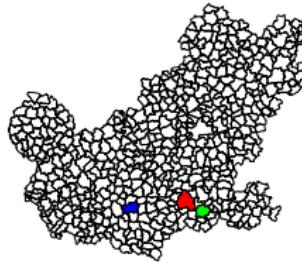
# Shapefiles zusammenfassen

```
vwb6c <- unionSpatialPolygons(vwb6,  
                                rep(1,length(vwb6)))  
plot(vwb6c,col="royalblue")
```



# Wo ist Mannheim?

```
Com <- vwb6@data$NAME  
plot(vwb6)  
plot(vwb6[Com=="Mannheim",], col="red", add=T)  
plot(vwb6[Com=="Heidelberg",], col="green", add=T)  
plot(vwb6[Com=="Kaiserslautern",], col="blue", add=T)
```



# Paket rgdal - PLZ Datensatz einlesen

## Quelle für PLZ Shapefiles

```
library(rgdal)

setwd(data_path)
plz <- readOGR ("post_pl.shp","post_pl")

## OGR data source with driver: ESRI Shapefile
## Source: "D:\Daten\GeoDaten\post_pl.shp", layer: "post"
## with 8270 features
## It has 3 fields
```

# Die Daten plotten

```
plzbereich <- substr(plz@data$PLZ99,1,2)  
plot(plz[plzbereich=="68",])
```



# Die Grenze von Mannheim

```
ma_map <- plz[plz$PLZORT99=="Mannheim",]  
plot(ma_map)
```



# Die PLZ-Bereiche von Mannheim zusammenfassen

- Wir nutzen den Befehl `unionSpatialPolygons` im Paket `maptools`

```
library(maptools)
ma_map2 <- unionSpatialPolygons(SpP = ma_map,
                                 IDs = rep(1,length(ma_map)))
plot(ma_map2)
```



# Global Administraive Boundaries - GADM - NUTS level 1

```
library(raster)
```

```
library(raster)
```

```
LUX1 <- getData('GADM', country='LUX', level=1)  
plot(LUX1)
```



# Ein Blick auf die Daten

Koordinaten im polygon slot

```
LUX1@polygons[[1]]@Polygons[[1]]@coords
```

```
##           [,1]      [,2]
## [1,] 6.026519 50.17767
## [2,] 6.031361 50.16563
## [3,] 6.035646 50.16410
## [4,] 6.042747 50.16157
## [5,] 6.043894 50.16116
## [6,] 6.048243 50.16008
```

# Der Datenslot

```
head(LUX1@data)
```

```
##   OBJECTID ID_0 ISO      NAME_0 ID_1      NAME_1 HASC_1 CCM
## 1           1 LUX Luxembourg 1 Diekirch LU.DI
## 2           2 LUX Luxembourg 2 Grevenmacher LU.GR
## 3           3 LUX Luxembourg 3 Luxembourg LU.LU
##   TYPE_1 ENGTTYPE_1 NL_NAME_1          VARNAME_1
## 1 District    District          Dikrech|Dikkrich
## 2 District    District          Gréivemaacher
## 3 District    District          Lëtzebuerg|Luxemburg
```

# GADM- NUTS level 3

```
LUX3 <- getData('GADM', country='LUX', level=3)  
plot(LUX3)
```



# GADM- NUTS level 4

```
LUX4 <- getData('GADM', country='LUX', level=4)  
plot(LUX4)
```



# GADM- NUTS level 3

```
DEU3 <- getData('GADM', country='DEU', level=3)  
plot(DEU3)
```



# Gemeinden in Deutschland

Bundesamt für Kartographie und Geodäsie (BKG)

```
library(maptools)
krs <- readShapePoly("vg250_krs.shp")
plot(krs)
```



# Kreise eines Bundeslandes

```
fds <- substr(krs@data$AGS, 1, 2)
```

```
plot(krs[fds=="05",])
```



# Andere Quellen

- World Port Index

```
library(rgdal)
WPI <- readOGR ("WPI.shp","WPI")
plot(WPI)
```



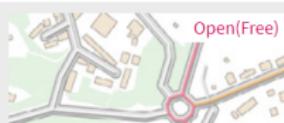
# Weitere Quellen für Shapefiles

- **Eurostat Karten** - in der Regel die Europäischen Mitgliedsstaaten
- **Open linked data** - Ordnance Survey (GB)



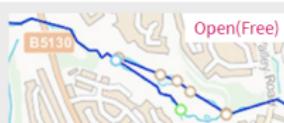
OS Names API ›

Addressing and Location



OS Open Roads ›

Networks



OS Open Rivers ›

Networks



OS Terrain 50 ›

Height

- **World Borders Datensatz**
- **National Historical Information System**
- **Freie Polygon-Daten für die USA**
- Überblick über - **Spatial Data in R**