

A4 - Das R-Paket tmap

Jan-Philipp Kolb

22 Oktober 2018

Inhalt dieses Abschnitts

- Das Paket `tmap` wird vorgestellt.
- Die Datenquelle `naturalearthdata` wird vorgestellt.
- Es wird gezeigt, wie man Karten von Europa, der Welt und einzelnen Ländern erzeugen kann

Das Paket tmap

Thematische Karten

- Mit dem Paket **tmap** kann man thematische Karten erzeugen
- Die folgenden Beispiele basieren auf der **Vignette** des Paketes .

```
install.packages("tmap")
```

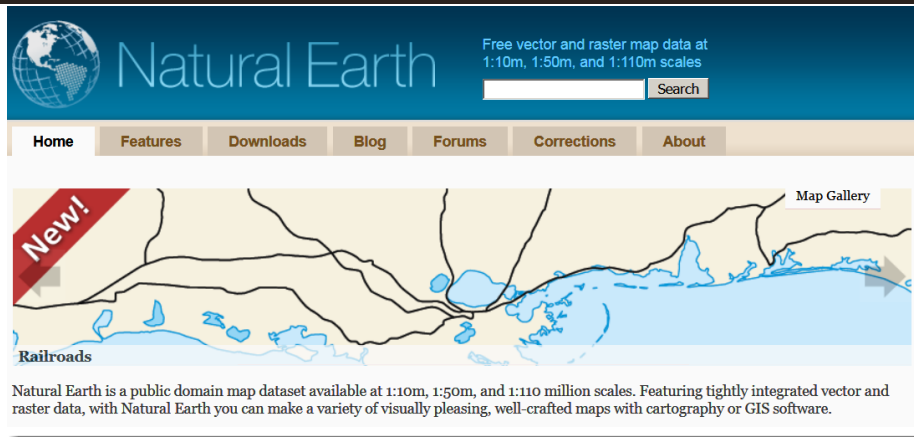
```
library(tmap)
```

Der Europe Datensatz

Natural Earth

- Datensatz enthält Informationen von **Natural Earth**

```
data(Europe)
```



The screenshot shows the Natural Earth website interface. At the top, there is a blue header with a globe icon on the left, the text "Natural Earth" in the center, and "Free vector and raster map data at 1:10m, 1:50m, and 1:110m scales" on the right. Below the header is a navigation bar with links: Home, Features, Downloads, Blog, Forums, Corrections, and About. The main content area features a map of Europe with a red "New!" banner in the top left corner. A label "Railroads" is visible on the map, and a "Map Gallery" link is in the top right. Below the map, a paragraph describes the dataset: "Natural Earth is a public domain map dataset available at 1:10m, 1:50m, and 1:110 million scales. Featuring tightly integrated vector and raster data, with Natural Earth you can make a variety of visually pleasing, well-crafted maps with cartography or GIS software."

Der Befehl `qtm` aus dem Paket `tmap`

Schnelle thematische Karte

- Mit dem Befehl **`qtm`** kann man eine schnelle thematische Karte erzeugen
- Beispiel aus der **Vignette** zum Paket `tmap`

```
qtm(Europe)
```



Der Europa-Datensatz

Der Europa Datensatz im Paket tmap

RPubs brought to you by RStudio

Show 10 entries

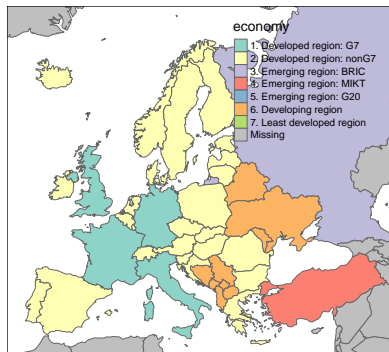
Search:

iso_a3	name	sovereign	continent	part	EU_Schengen	area	pop_est	pop_est_dens	gdp_md_est	gdp_cap_est	economy
ALB	Albania	Albania	Europe	Southern Europe		27400	3639453	132.826751824818	21810	5992.65878691111	6. Developing region
ALA	Aland	Finland	Europe	Northern Europe		674.381272941594	27153	40.2635735739827	1563	57562.7002541156	2. Developed region: nonG7
AND	Andorra	Andorra	Europe	Southern Europe		470	83888	178.485106382979	3660	43629.6013732596	2. Developed region: nonG7
ARM	Armenia	Armenia	Asia								
AUT	Austria	Austria	Europe	Western Europe	EU Schengen	82409	8210281	99.6284507760075	329500	40132.6093467446	2. Developed region: nonG7
AZE	Azerbaijan	Azerbaijan	Asia								
BEL	Belgium	Belgium	Europe	Western Europe	EU Schengen	30280	10414336	343.934478203435	389300	37381.1638111158	2. Developed region: nonG7

Um mehr Farbe in die Karte zu bekommen

Entwicklungsstand der Wirtschaft

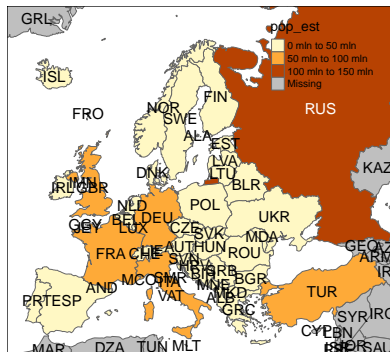
```
qtm(Europe, fill="economy")
```



Eine Karte mit Text

Bevölkerung

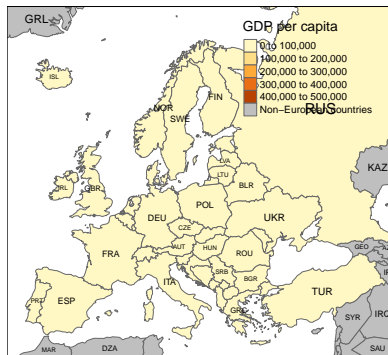
```
gtm(Europe, fill="pop_est", text="iso_a3")
```



Dieses Schema passt besser:

GDP

```
qtm(Europe, fill="gdp_cap_est", text="iso_a3",  
    text.size="AREA", root=5, fill.title="GDP per capita",  
    fill.textNA="Non-European countries", theme="Europe")
```



Themen des Europa-Datensatzes

Verfügbare Variablen im Datensatz

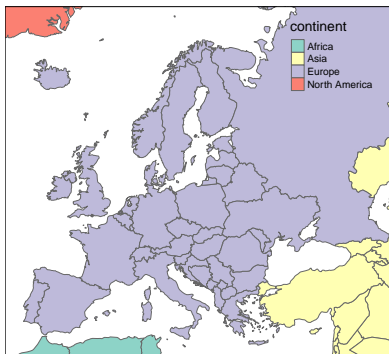
- **ISO Klassifikation**
- Ländername
- Ist das Land Teil Europas?
- Fläche, Bevölkerung, Bevölkerungsdichte,
- **Bruttoinlandsprodukt**
- Bruttoinlandsprodukt **zu Kaufkraftparitäten**
- Ökonomie, Einkommensgruppe

Der Europa Datensatz - Variablen und was dahinter steckt

	iso_a3	name	sovereight	continent	part
5	ALB	Albania	Albania	Europe	Southern Europe
6	ALA	Aland	Finland	Europe	Northern Europe
7	AND	Andorra	Andorra	Europe	Southern Europe
10	ARM	Armenia	Armenia	Asia	NA
17	AUT	Austria	Austria	Europe	Western Europe
18	AZE	Azerbaijan	Azerbaijan	Asia	NA
20	BEL	Belgium	Belgium	Europe	Western Europe
24	BGR	Bulgaria	Bulgaria	Europe	Eastern Europe

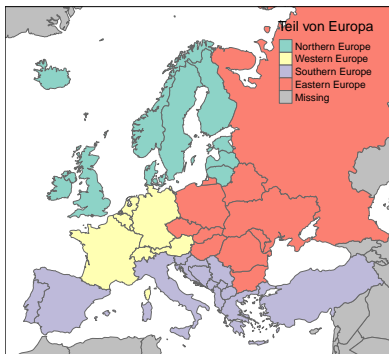
Die Variable continent

```
qtm(Europe, fill="continent")
```



Die Variable part

```
qtm(Europe, fill="part",fill.title="Teil von Europa")
```



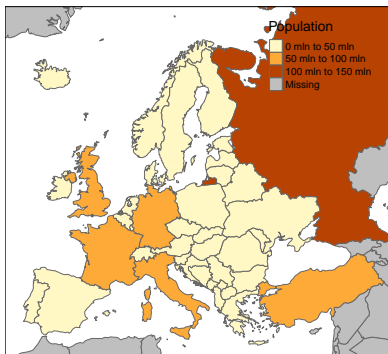
Die Variable area

```
qtm(Europe, fill="area") # Russia is huge
```

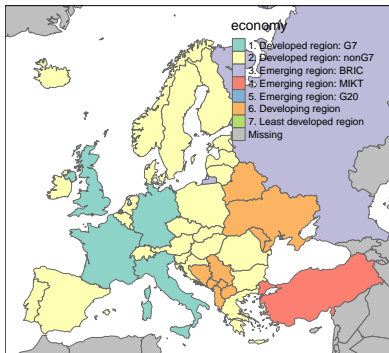


Bevölkerung

```
qtm(Europe, fill="pop_est", fill.title="Population")
```

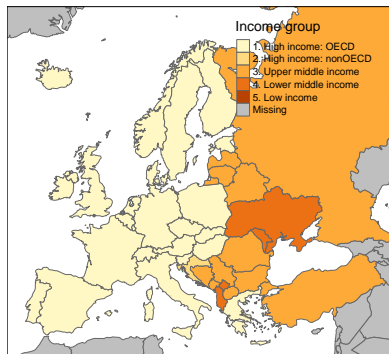


```
qtm(Europe, fill="economy")
```



Einkommensgruppe

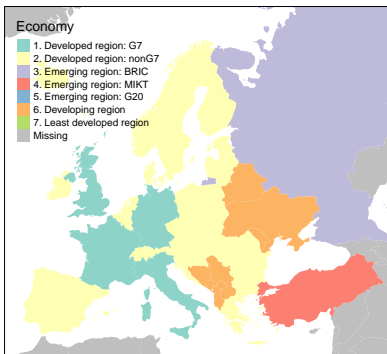
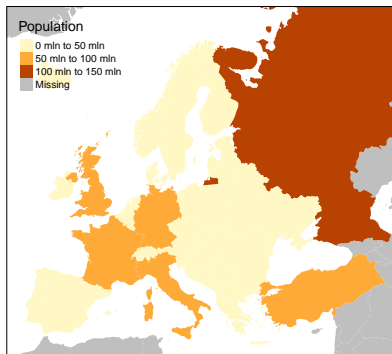
```
qtm(Europe, fill="income_grp",fill.title="Income group")
```



Zwei Karten

Bevölkerung und Entwicklungsstand

```
tm_shape(Europe) +  
  tm_fill(c("pop_est", "economy"),  
    title=c("Population", "Economy"))
```



Der Datensatz World im Paket tmap

Ähnlich wie der Europe Datensatz nur für die ganze Welt

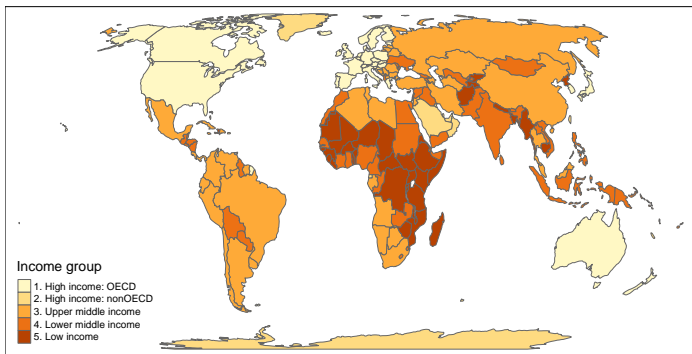
```
data(World)
```

RPubs brought to you by RStudio

Show 10 entries		Search: <input type="text"/>													
	iso_a3	name	sovereign	continent	subregion	area	pop_est	pop_est_dens	gdp_md_est	gdp_cap_est	economy	income_grp	life_exp	well_being	HPI
2	AFG	Afghanistan	Afghanistan	Asia	Southern Asia	652860	28400000	43.5009037159575	22270	784.154929577465	7. Least developed region	5. Low income	48.7	4.75838085759722	36.753657778004
3	AGO	Angola	Angola	Africa	Middle Africa	1246700	12799293	10.2665380604797	110300	8617.6634912569	7. Least developed region	3. Upper middle income	51.1	4.20609164016618	33.2014320444336
5	ALB	Albania	Albania	Europe	Southern Europe	27400	3639453	132.826751824818	21810	5992.65878691111	6. Developing region	4. Lower middle income	76.9	5.26893660419411	54.051180370208
8	ARE	United Arab Emirates	United Arab Emirates	Asia	Western Asia	83600	4798491	57.3982177033493	184300	38407.907819154	6. Developing region	2. High income: notOECD	76.5	7.19680309333638	31.778274185231
9	ARG	Argentina	Argentina	South America	South America	2736690	40913584	14.9500250302373	573900	14027.1260518267	5. Emerging region: G20	3. Upper middle income	75.9	6.44106720496824	54.0550416711541
10	ARM	Armenia	Armenia	Asia	Western Asia	28470	2967004	104.215103617843	18770	6326.24694809983	6. Developing region	4. Lower middle income	74.2	4.36781129220333	46.0031857989857
12	ATA	Antarctica	Antarctica	Antarctica	Antarctica	10866664.4069415	3802	0.00034987733730556			6. Developing region	2. High income: notOECD			
14	ATF	Fr. S. Antarctic Lands	France	Seven seas (open ocean)	Seven seas (open ocean)	6187.20529564552	140	0.0226273403435523	16	114285.714285714	6. Developing region	2. High income: notOECD			
16	AUS	Australia	Australia	Oceania	Australia and New Zealand	7682300	21262641	2.76774416515887	800200	37634.0831790369	2. Developed region: nonG7	1. High income: OECD	81.9	7.40561614869191	41.9798119494163
17	AUT	Austria	Austria	Europe	Western Europe	82409	8210281	99.6284507760075	329500	40132.6093467446	2. Developed region: nonG7	1. High income: OECD	80.9	7.34603595780621	47.0851352018778

Welt - Länder nach Einkommensgruppe

```
qtm(World, fill="income_grp",fill.title="Income group")
```

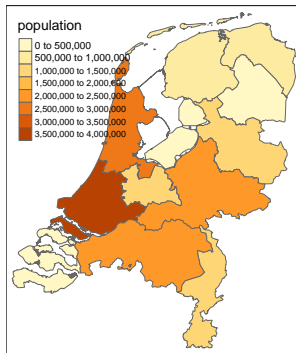


Ein Datensatz zu den Provinzen in den Niederlanden (R-Paket tmap)

	code	name	population	pop_men	pop_women	pop_0_14
0	20	Groningen	582705	289795	292875	15
1	21	Friesland	646290	323215	323055	17
2	22	Drenthe	488970	242225	246755	17
3	23	Overijssel	1139680	570185	569465	18
4	24	Flevoland	399885	199940	199940	20
5	25	Gelderland	2019635	997805	1021790	17

Niederlande - Bevölkerung in den Provinzen

```
qtm(NLD_prov, fill="population", fill.title="population")
```



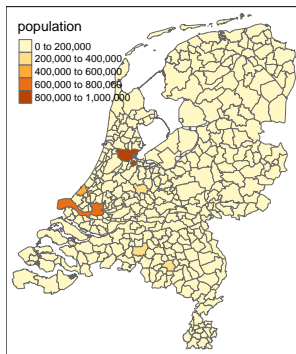
Ein Datensatz zu den Gemeinden in den Niederlanden

```
data(NLD_muni)
```

	name	province	population
0	Appingedam	Groningen	12065
1	Bedum	Groningen	10495
2	Bellingwedde	Groningen	8920
3	Ten Boer	Groningen	7480
4	Delfzijl	Groningen	25695
5	Groningen	Groningen	198315
6	Grootegast	Groningen	12165
7	Haren	Groningen	18780
8	Hoogezand-Sappemeer	Groningen	34305
9	Leek	Groningen	19595
10	Loppersum	Groningen	10195
11	Marum	Groningen	10375

Bevölkerung der Gemeinden in den Niederlanden

```
qtm(NLD_muni, fill="population")
```



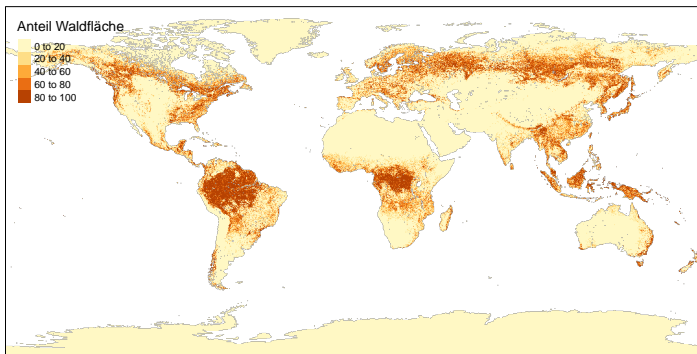
Räumliche Daten zur Flächennutzung

```
data(land)
data(World)
```

	cover_cls	trees
321215	Water	NA
23639	Water	NA
188899	Water	NA
460085	Water	NA
434085	Water	NA
355258	Water	NA
475540	Water	NA
508147	Water	NA
174903	Water	NA
574241	Snow/ice	0

Weltweite Flächennutzung

```
tm_shape(land, relative=FALSE) +  
  tm_raster("trees", title="Anteil Waldfläche")
```



Räumliche Daten zu Metropolregionen

UN - World Urbanization Prospects 2018

```
data(metro)
```

Show entries

Search:

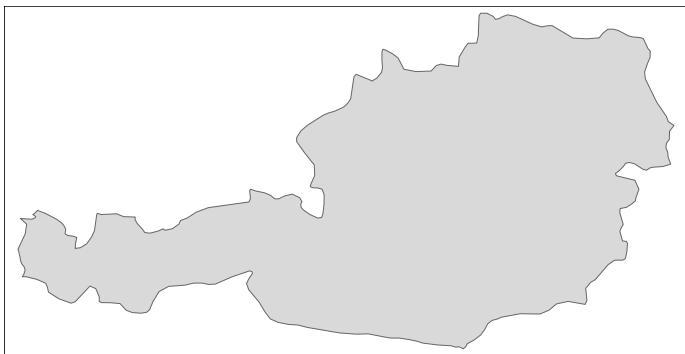
	name	name_long	iso_a3	pop1950	pop1960	pop1970	pop1980	pop1990	pop2000	pop2010	pop2020	pop2030
2	Kabul	Kabul	AFG	170784	285352	471891	977824	1549320	2401109	3722320	5721697	8279607
8	Algiers	El Djazair (Algiers)	DZA	516450	871636	1281127	1621442	1797068	2140577	2432023	2835218	3404575
13	Luanda	Luanda	AGO	138413	219427	459225	771349	1390240	2591388	4508434	6836849	10428756
16	Buenos Aires	Buenos Aires	ARG	5097612	6597634	8104621	9422362	10513284	12406780	14245871	15894307	16956491
17	Cordoba	Cordoba	ARG	429249	605309	809794	1009521	1200168	1347561	1459268	1562509	1718192
25	Rosario	Rosario	ARG	554483	671349	816230	953491	1083819	1152387	1298073	1453814	1606993
32	Yerevan	Yerevan	ARM	341432	537759	778158	1041587	1174524	1111301	1065597	1023703	1057459
33	Adelaide	Adelaide	AUS	429277	571822	850168	971856	1081618	1141623	1217990	1320783	1505422
34	Brisbane	Brisbane	AUS	441718	602999	904777	1134833	1381306	1666203	2033617	2388517	2721325
37	Melbourne	Melbourne	AUS	1331966	1851220	2499109	2839019	3154314	3460541	3951216	4500501	5070873

Showing 1 to 10 of 436 entries

Previous 2 3 4 5 ... 44 Next

Nur ein Land visualisieren

```
tm_shape(Europe[Europe$name=="Austria", ]) +  
  tm_polygons()
```



Beispieldaten laden

Datenquelle Eurostat

- Daten zur Arbeitslosigkeit in Europa

```
url <- "https://raw.githubusercontent.com/Japhilko/  
GeoData/master/2015/data/Unemployment07a13.csv"
```

```
Unemp <- read.csv(url)
```

Überblick über die Daten

X	GEO	Val2007M12	Val2013M01
9316	EU28	6.9	10.9
9325	EU27	6.9	10.9
9334	EU25	6.9	11.0
9343	EU15	6.9	11.1
9352	EA	7.3	12.0
9361	EA19	7.3	12.0
9370	EA18	7.4	12.0
9379	EA17	7.4	12.0
9388	EA16	7.4	12.0
9397	EA15	7.3	12.0

Exkurs: der Befehl match

Zwei Beispielvektoren erstellen:

```
vec_a <- c("A",2,6,1,"C")  
vec_b <- c(1,"C",2)
```

Die beiden Vektoren zusammen bringen

- Mit der Funktion `match` kann man schauen, welches Element des ersten Vektors sich im zweiten Vektor wiederfindet.

```
match(vec_a,vec_b)
```

```
## [1] NA  3 NA  1  2
```

Nutzung des Paketes tmap mit eigenen Daten

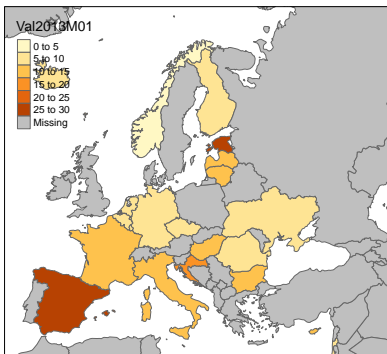
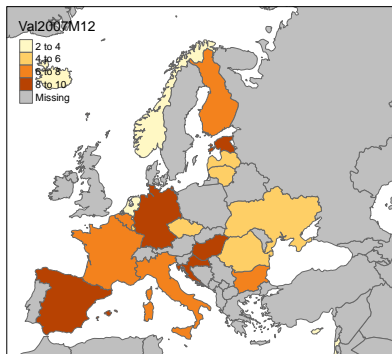
```
library("tmap")  
data(Europe)
```

Die Daten matchen

```
iso_a2<- substr(Europe@data$iso_a3,1,2)  
ind <- match(iso_a2,Unemp$GEO)  
Europe@data$Val2007M12 <- Unemp$Val2007M12[ind]  
Europe@data$Val2013M01 <- Unemp$Val2013M01[ind]
```


Eine Karte erzeugen

```
qtm(Europe, c("Val2007M12", "Val2013M01"))
```



A4A Übung: Visualisierung von Eurostat Daten

- Verbinde die Statistik zur Sparquote mit den Kartendaten.
- Stelle die Daten in einer Karte dar.

Kleine und viele Karten

```
tm_shape(Europe[Europe$continent=="Europe",]) +  
  tm_fill("part", thres.poly = 0) +  
  tm_facets("name", free.coords=TRUE)
```



Das Paket tmap zitieren

```
citation("tmap")
```

```
##  
## To cite tmap/tmaptools in publications use:  
##  
## Tennekes M (2018). "tmap: Thematic Maps in R." _Journal of  
## Statistical Software_, 84(6), 1-39. doi: 10.18637/jss.v084.i06  
## (URL: http://doi.org/10.18637/jss.v084.i06).  
##  
## A BibTeX entry for LaTeX users is  
##  
## @Article{,  
##   title = {{tmap}: Thematic Maps in {R}},  
##   author = {Martijn Tennekes},  
##   journal = {Journal of Statistical Software},  
##   year = {2018},  
##   volume = {84},  
##   number = {6},  
##   pages = {1-39},  
##   doi = {10.18637/jss.v084.i06},  
##   url = {http://doi.org/10.18637/jss.v084.i06},  
## }
```