

# R-package tmap

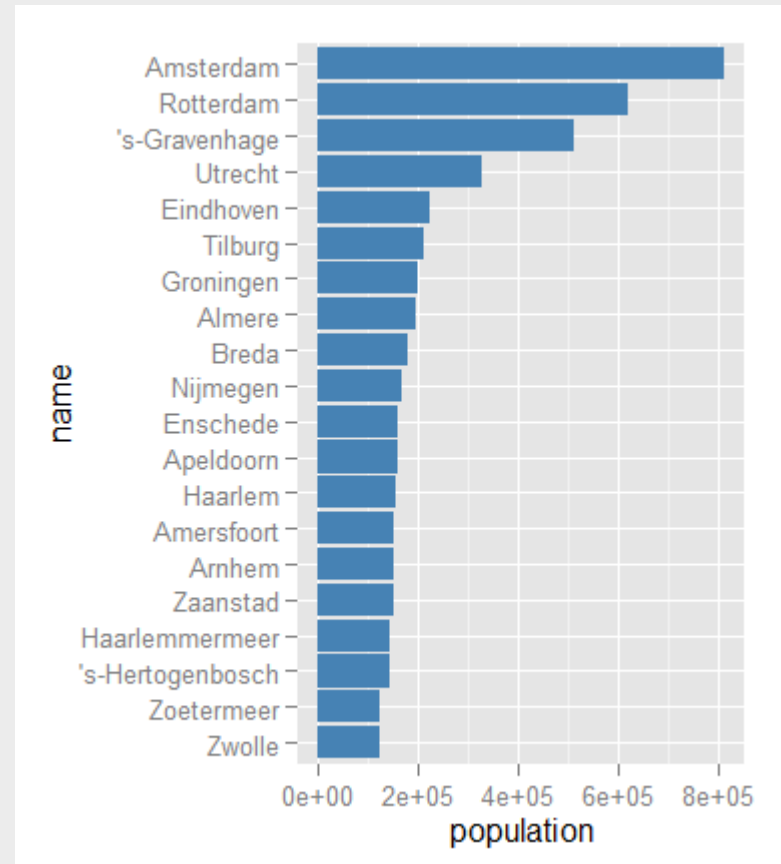
Creating thematic maps in a flexible way

**Martijn Tennekes**



Statistics  
Netherlands

# Thematic map



Geographic map

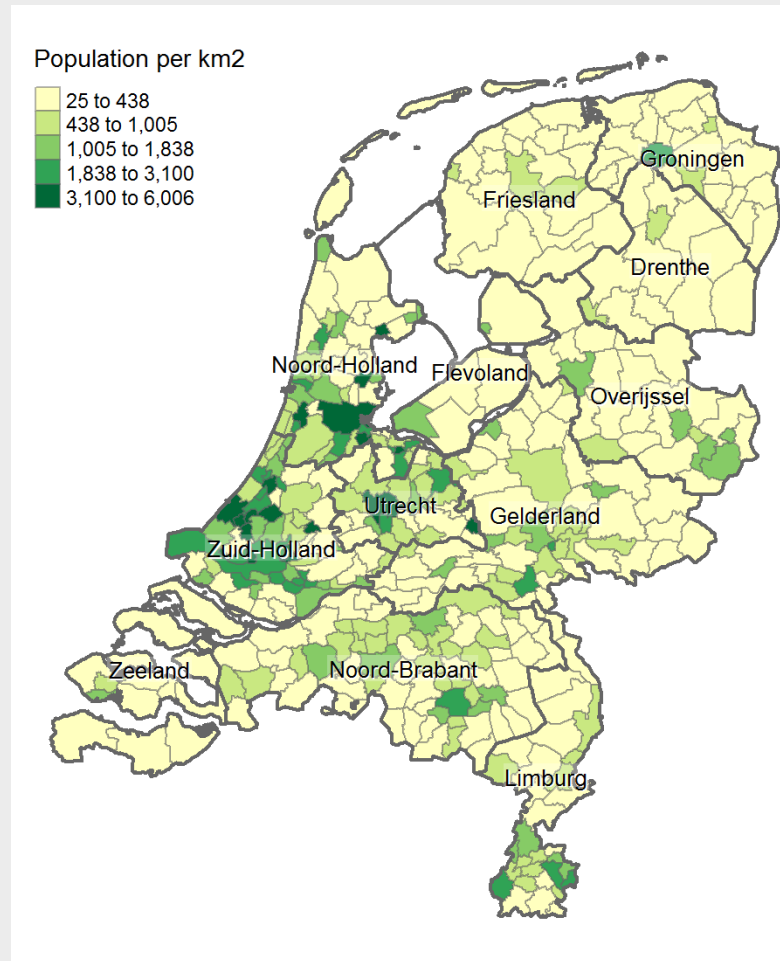
+

Theme

2



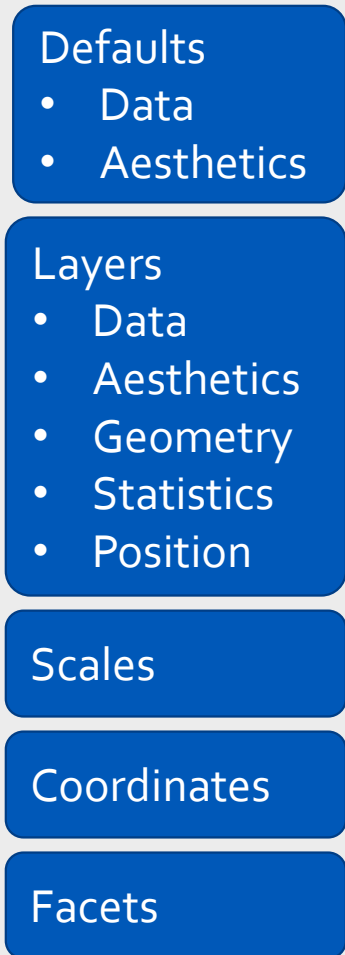
# Thematic map



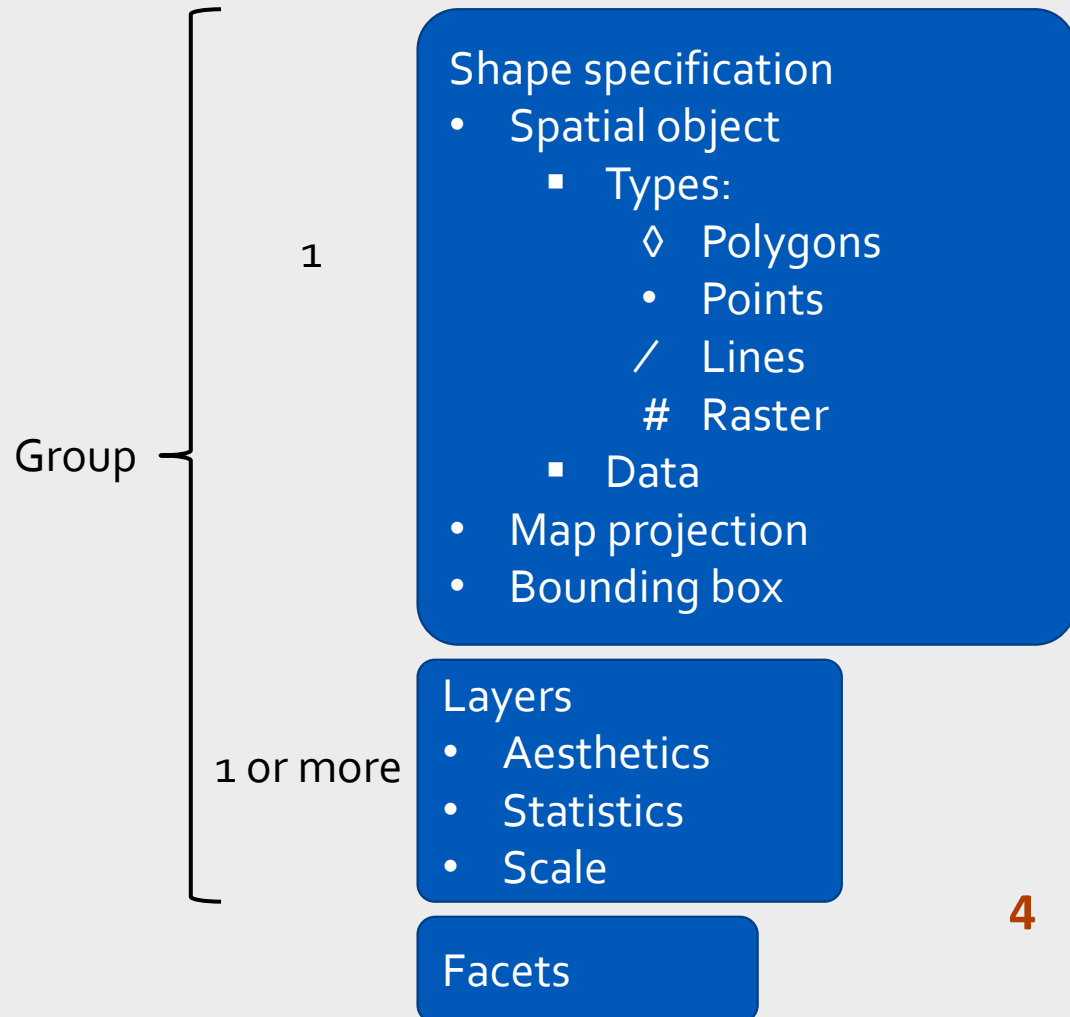
= Thematic map

# Layered approach

A Layered Grammar of Graphics (Wickham, 2010)  
Implemented in **ggplot2**



Layered approach in **tmap**



# Building a thematic map

```
tm_shape(NLD_muni,  
         projection="rd") +
```

```
tm_fill()
```



# Building a thematic map

```
tm_shape(NLD_muni,  
         projection="rd") +
```

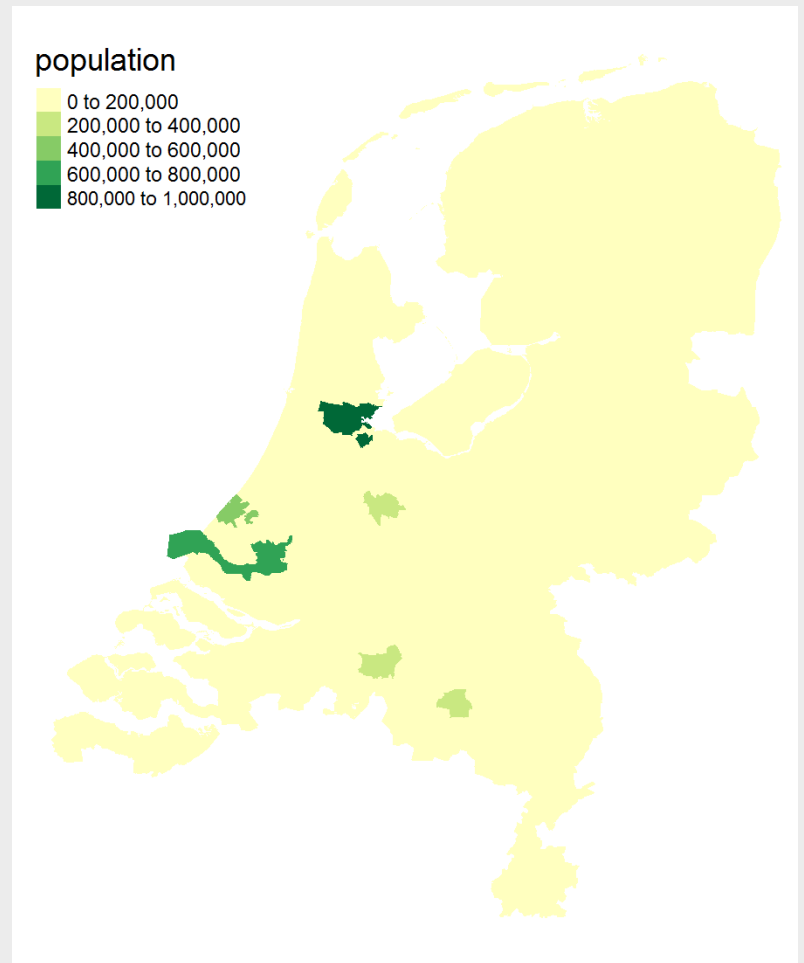
```
tm_fill("blue")
```



# Building a thematic map

```
tm_shape(NLD_muni,  
         projection="rd") +
```

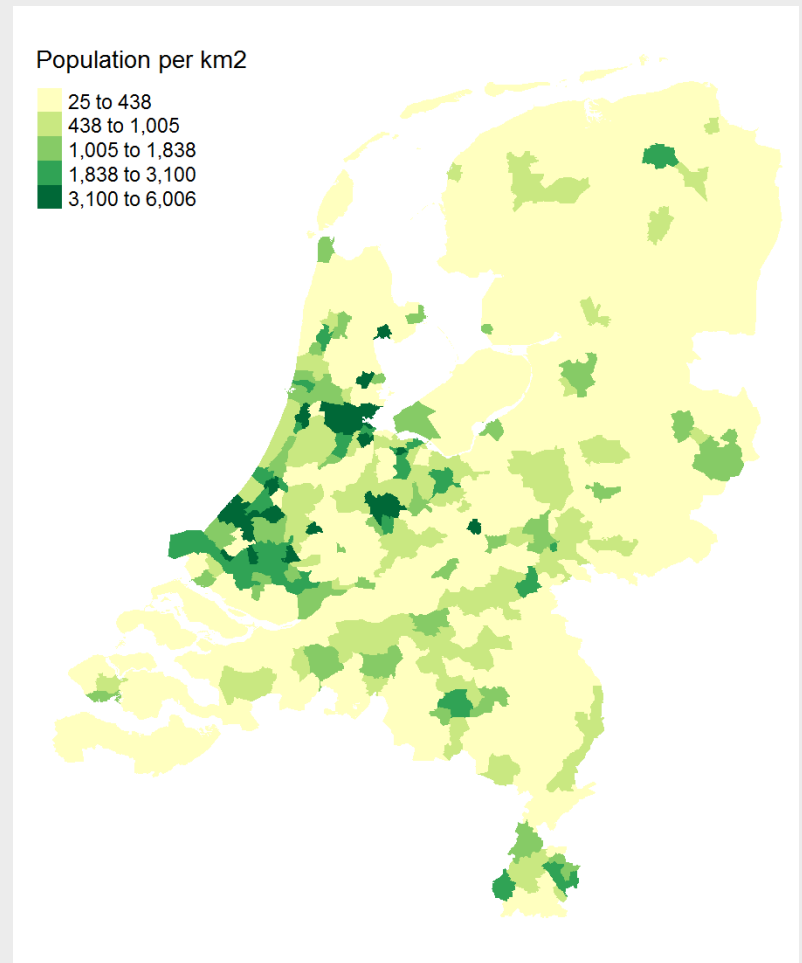
```
tm_fill("population")
```



# Building a thematic map

```
tm_shape(NLD_muni,  
         projection="rd") +
```

```
tm_fill("population",  
       convert2density=TRUE,  
       style="kmeans",  
       title="Population per km2") +
```



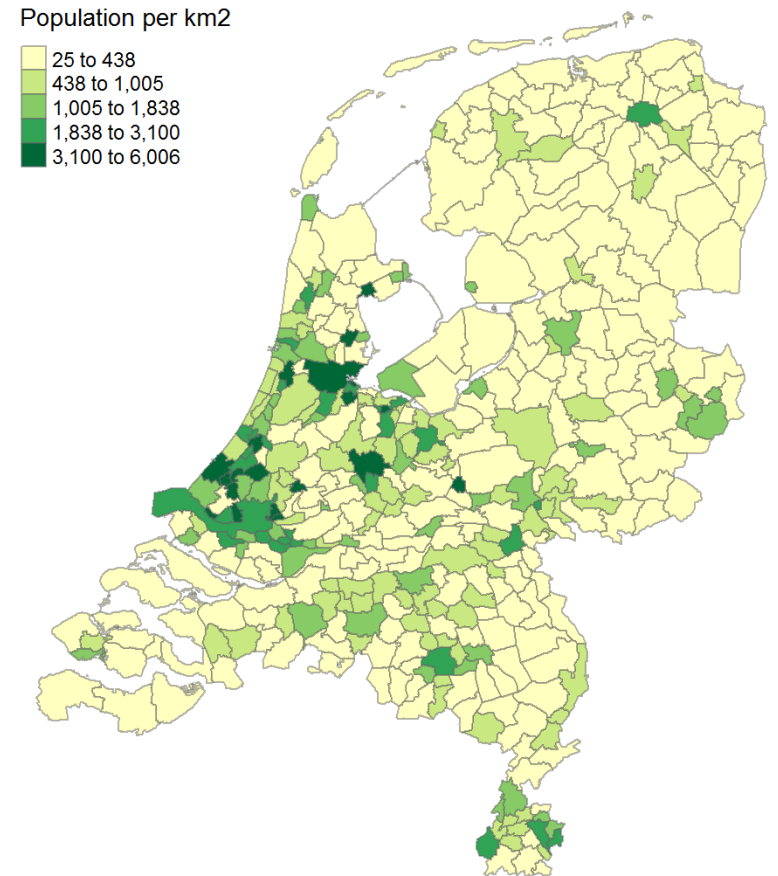


# Building a thematic map

```
tm_shape(NLD_muni,  
         projection="rd") +
```

```
tm_fill("population",  
        convert2density=TRUE,  
        style="kmeans",  
        title="Population per km2") +
```

```
tm_borders(alpha=.5) +
```



# Building a thematic map

```
tm_shape(NLD_muni,  
         projection="rd") +
```

```
tm_fill("population",  
       convert2density=TRUE,  
       style="kmeans",  
       title="Population per km2") +
```

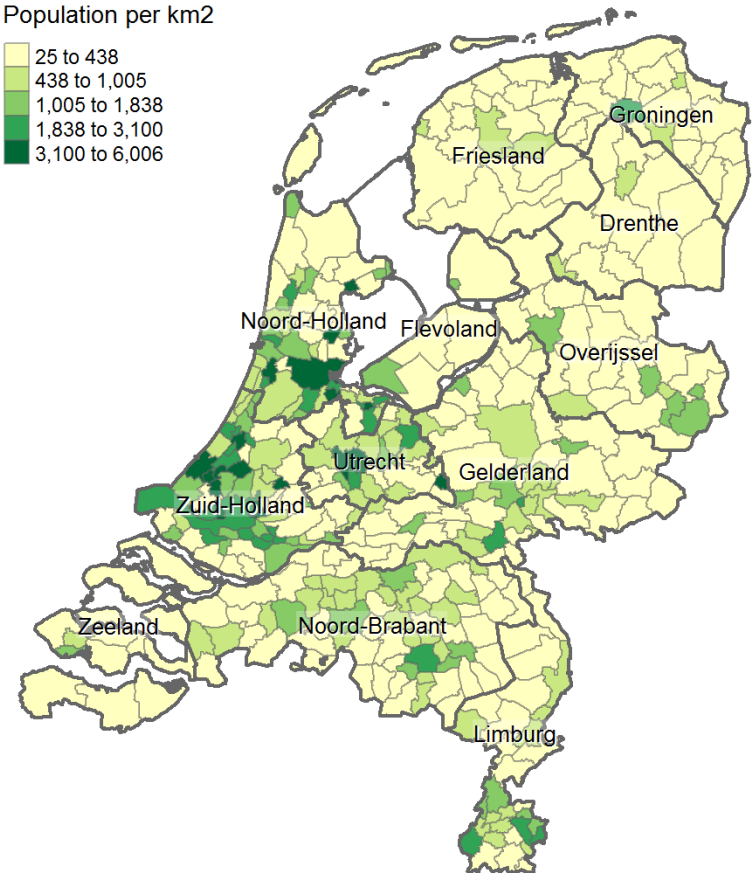
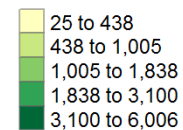
```
tm_borders(alpha=.5) +
```

```
tm_shape(NLD_prov) +
```

```
tm_borders(lwd=2) +
```

```
tm_text("name", size=.8, shadow=TRUE,  
       bg.color="white", bg.alpha=.25)
```

Population per km2

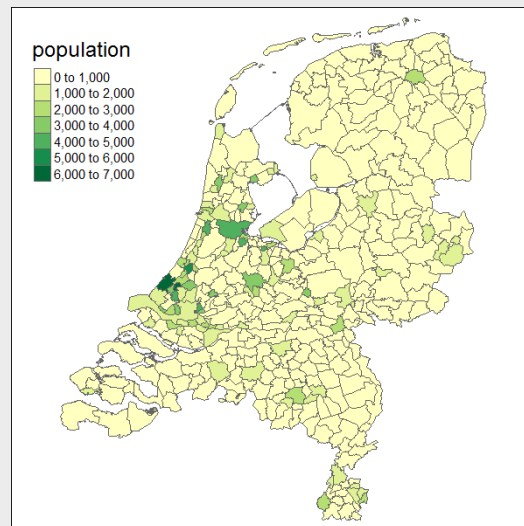


# Quick thematic map

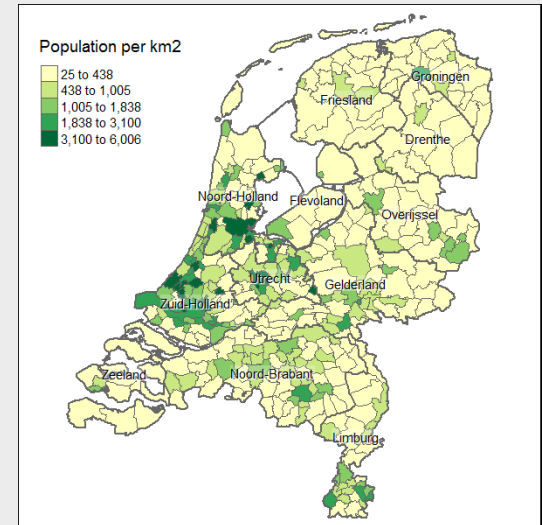
- Quick thematic map: qtm
- Wrapper for the main plotting method



```
qtm(NLD_muni)
```



```
qtm(NLD_muni,  
fill="population",  
convert2density=TRUE)
```

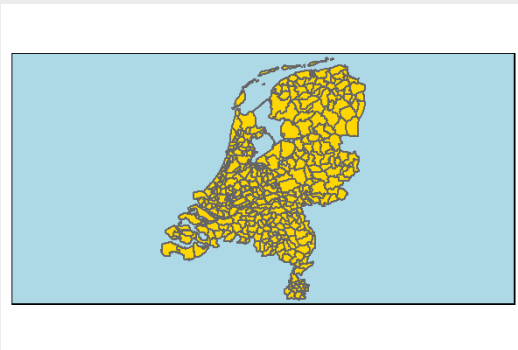


```
qtm(NLD_muni,  
fill="population",  
convert2density=TRUE,  
fill.style="kmeans",  
fill.title="Population per km2") +  
qtm(NLD_prov, fill=NULL,  
text="name", text.size=.7,  
borders.lwd=2,  
text.bg.color="white",  
text.bg.alpha=.25, shadow=TRUE)
```

# Aspect ratio and margins

Aspect ratio (=width/height) of the frame determined by:

user



`tm_layout(asp=2)`

device (and outer margins)



`tm_layout(asp=0)`

shape (and inner margins)



`tm_layout(asp=NA)`



Bounding box shape

Inner margins

Outer margins

`tm_layout(design.mode=TRUE)`

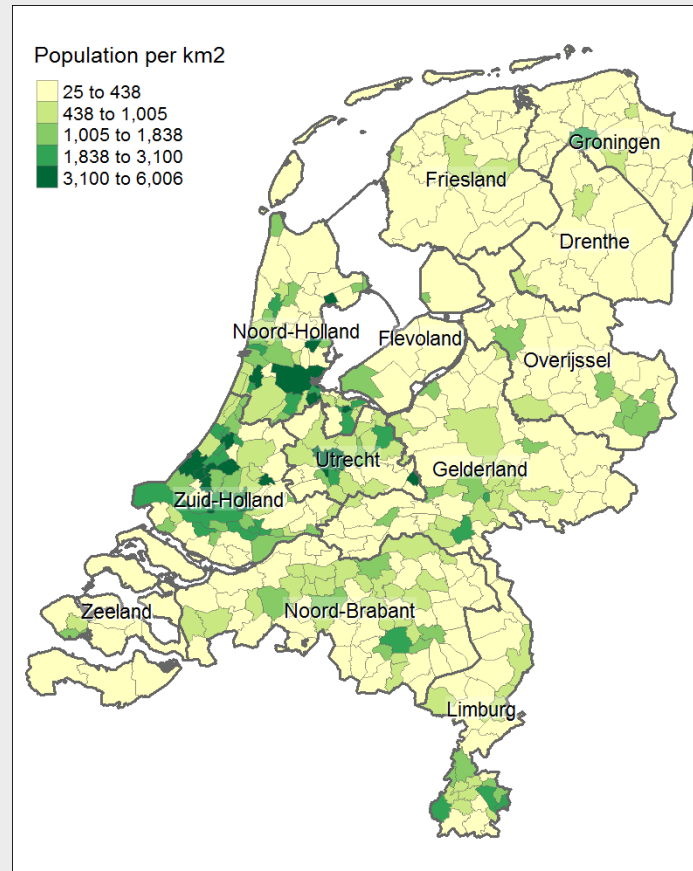
# Exporting thematic maps

```
tm <- tm_shape(NLD_prov) + ...  
  
png("map.png", width=800, height=1000)  
  tm + tm_layout(outer.margins=0, asp=0, scale=.8)  
dev.off()
```

Global scaling parameter

Frame fits perfectly  
in png image

map.png



# Exporting thematic maps

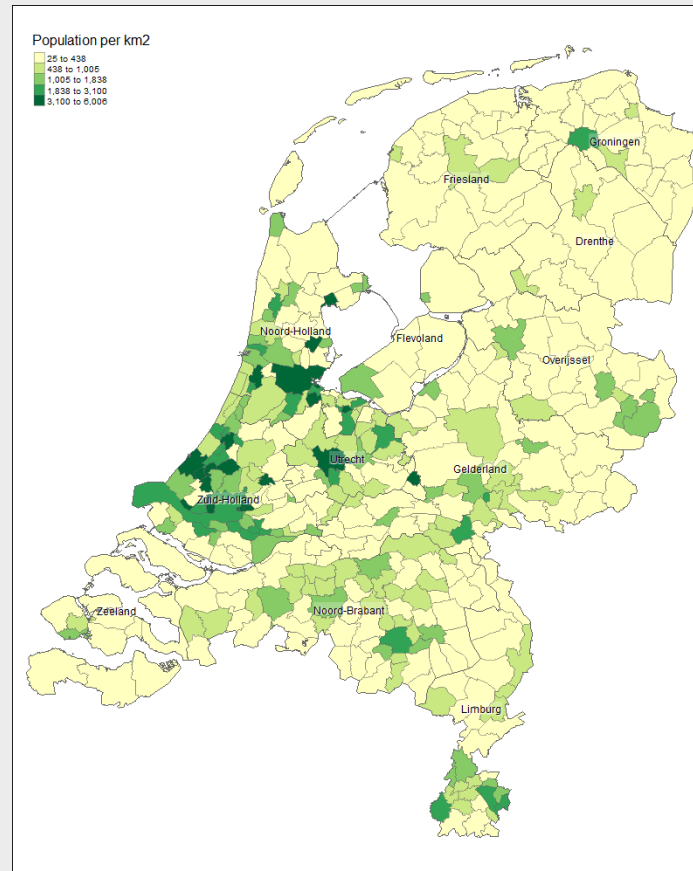
```
tm <- tm_shape(NLD_prov) + ...
```

```
png("map.png", width=800, height=1000)  
  tm + tm_layout(outer.margins=0, asp=0, scale=.4)  
dev.off()
```

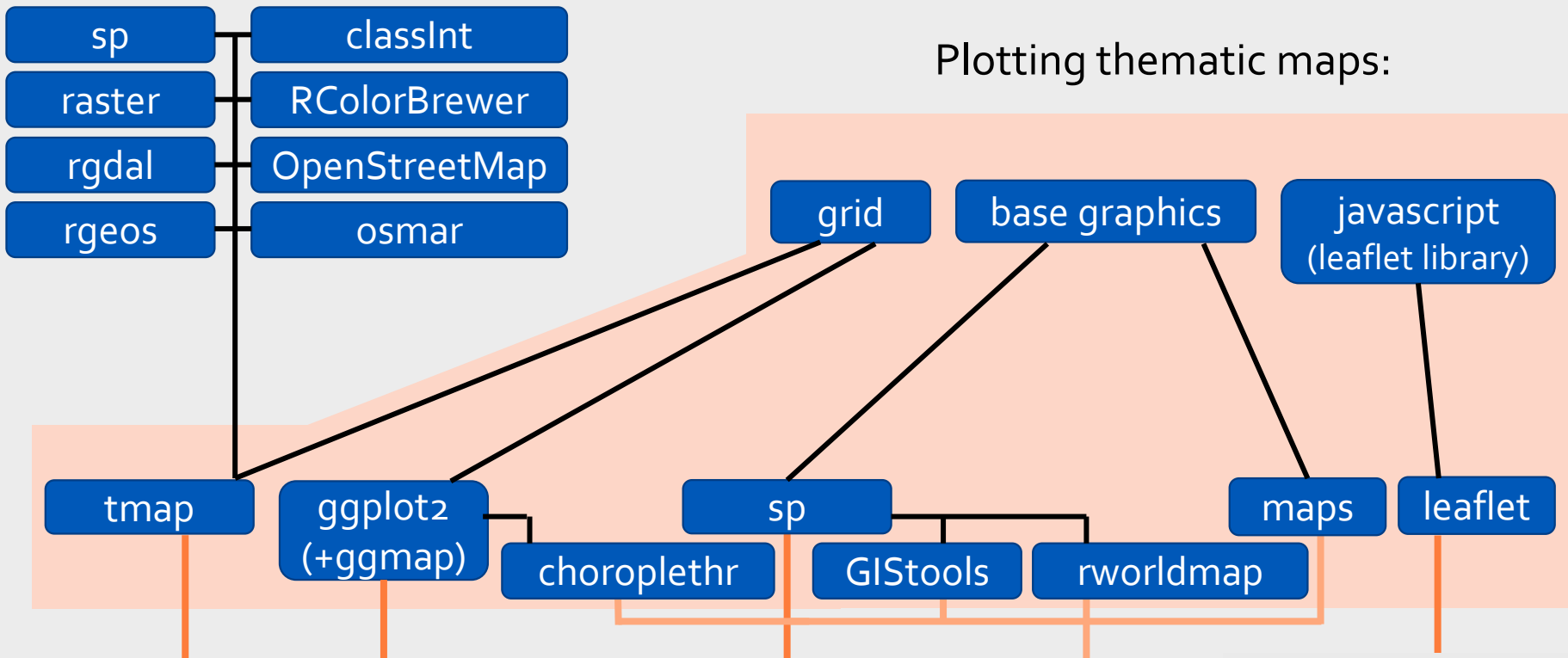
Global scaling parameter

Frame fits perfectly  
in png image

map.png



# tmap and the field



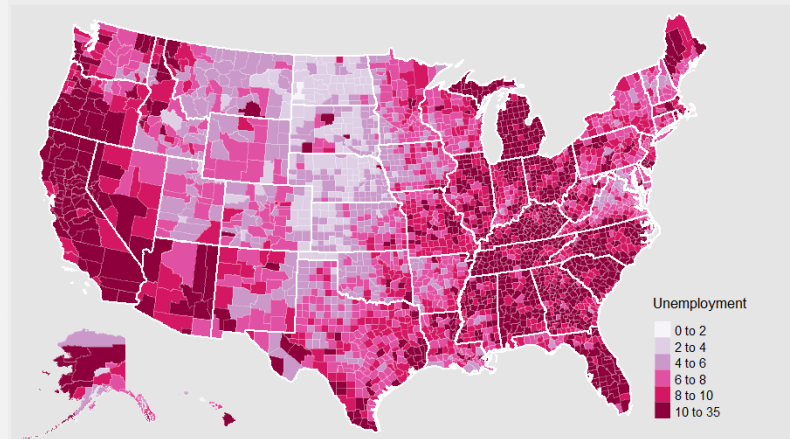
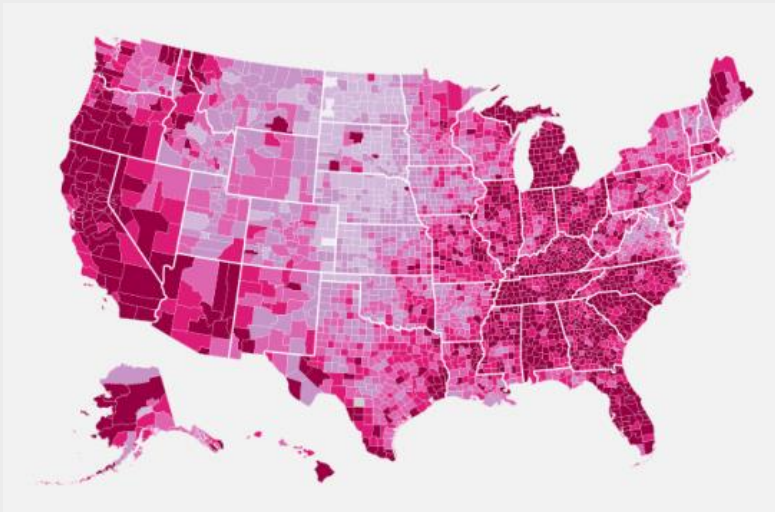
Plotting thematic maps:

- |   |  |  |  |   |
|---|--|--|--|---|
| <ul style="list-style-type: none"> <li>+ Easy to use</li> <li>+ Flexible</li> <li>+ Layer based</li> <li>+ OSM</li> <li>+ Small multiples</li> <li>- New syntax</li> <li>- Not interactive (yet)</li> </ul> | <ul style="list-style-type: none"> <li>+ Grammar of graphics</li> <li>+ Familiar syntax</li> <li>- Processing required:</li> <li>- Shape to be fortified</li> <li>- Layout to be polished</li> </ul> | <ul style="list-style-type: none"> <li>+ Familiar syntax</li> <li>- Do-it-yourself!</li> </ul> | <ul style="list-style-type: none"> <li>+ Less DIY work</li> <li>- New syntax</li> <li>- Limited possibilities</li> </ul> | <ul style="list-style-type: none"> <li>+ Interactive</li> <li>+ Flexible</li> <li>+ Layered based</li> <li>+ OSM</li> <li>- Small multiples</li> <li>- New syntax</li> <li>* Lower level (w.r.t. tmap)</li> </ul> |
|---|--|--|--|---|

# Choropleth 2009 challenge

<http://blog.revolutionanalytics.com/2009/11/choropleth-map-r-challenge.html>

<http://blog.revolutionanalytics.com/2009/11/choropleth-challenge-result.html>



Goal: recreate this map

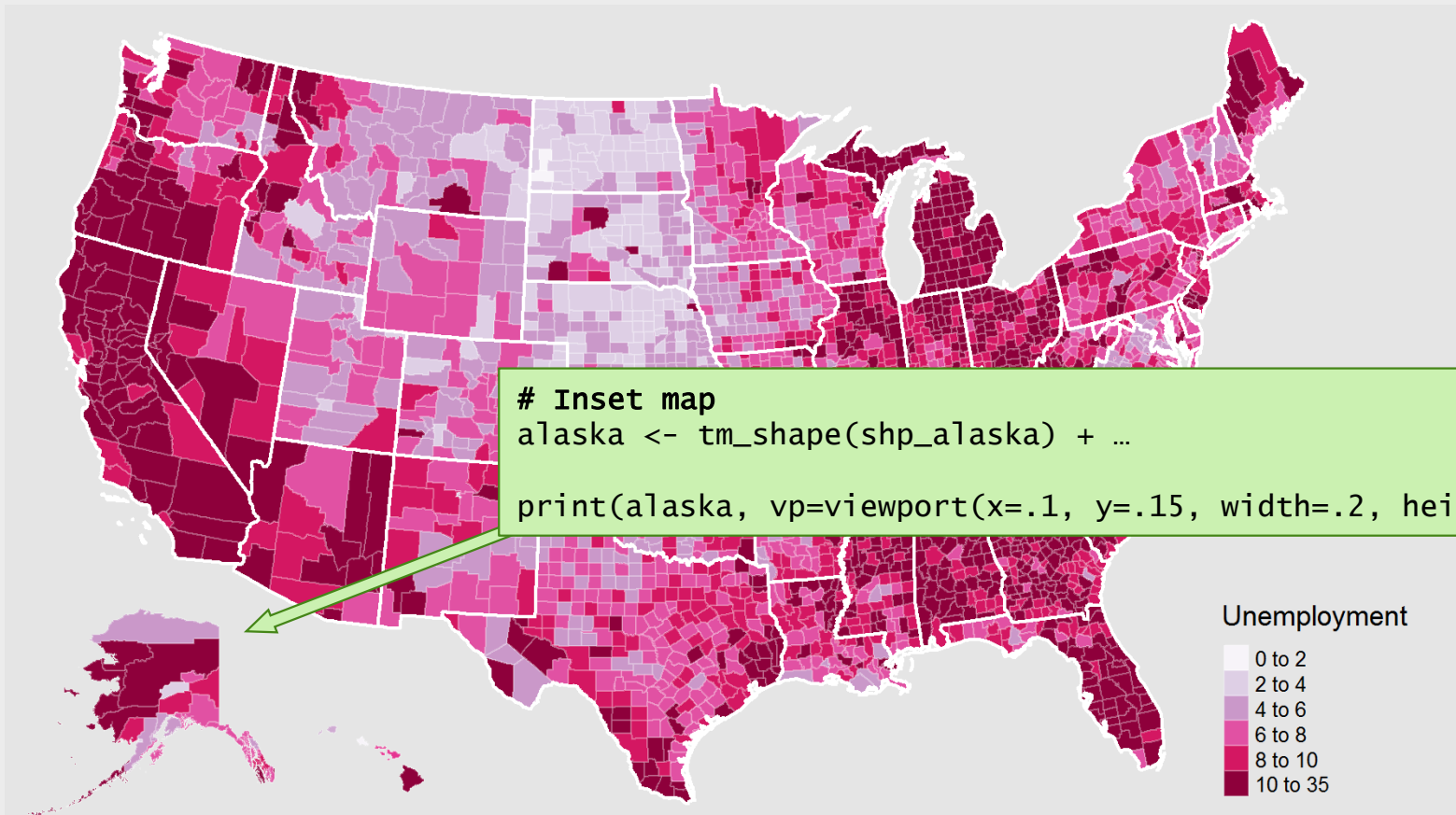
Result made with tmap



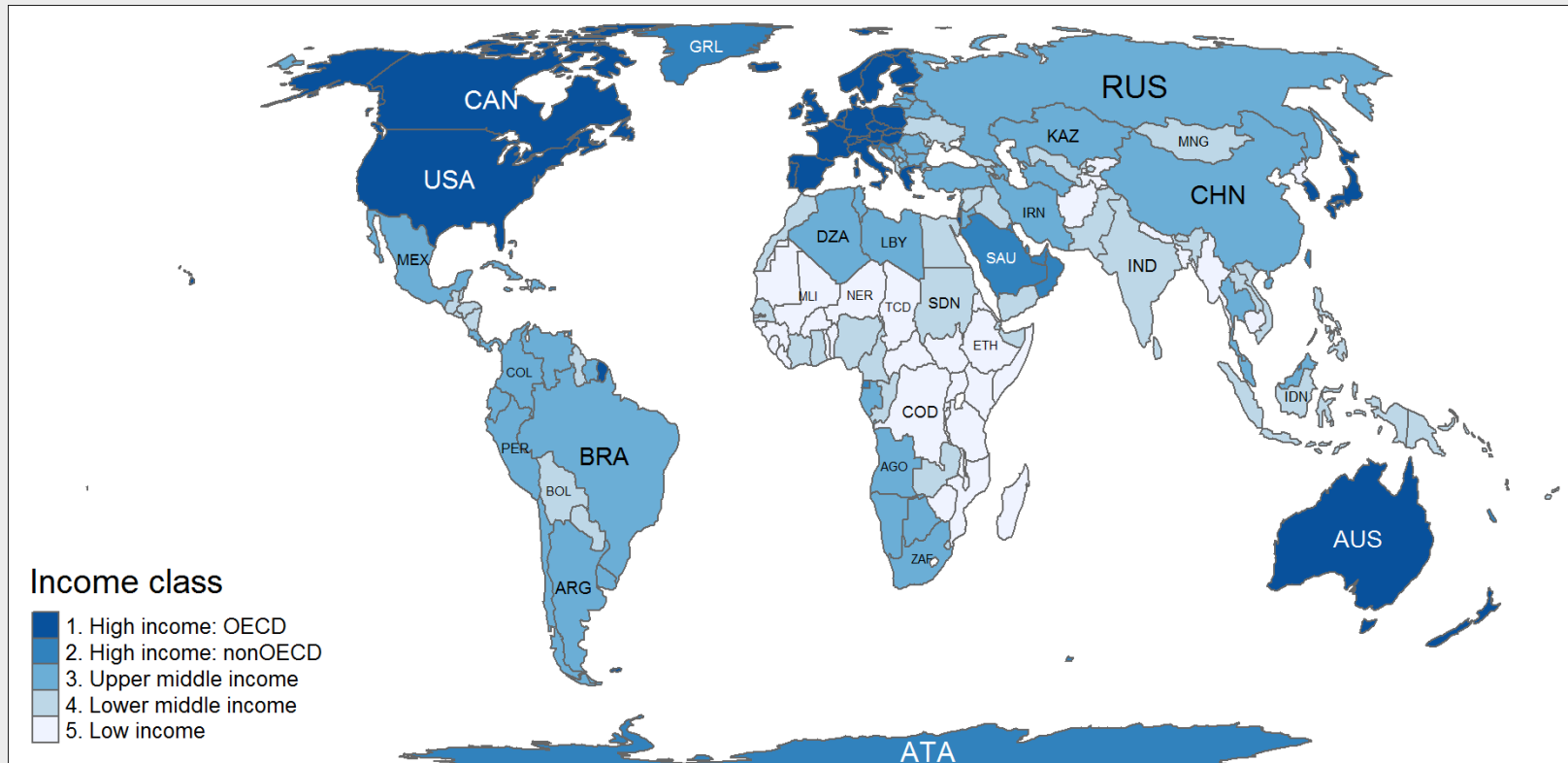
# Choropleth 2009 challenge

<http://blog.revolutionanalytics.com/2009/11/choropleth-map-r-challenge.html>

<http://blog.revolutionanalytics.com/2009/11/choropleth-challenge-result.html>



# Example: choropleth



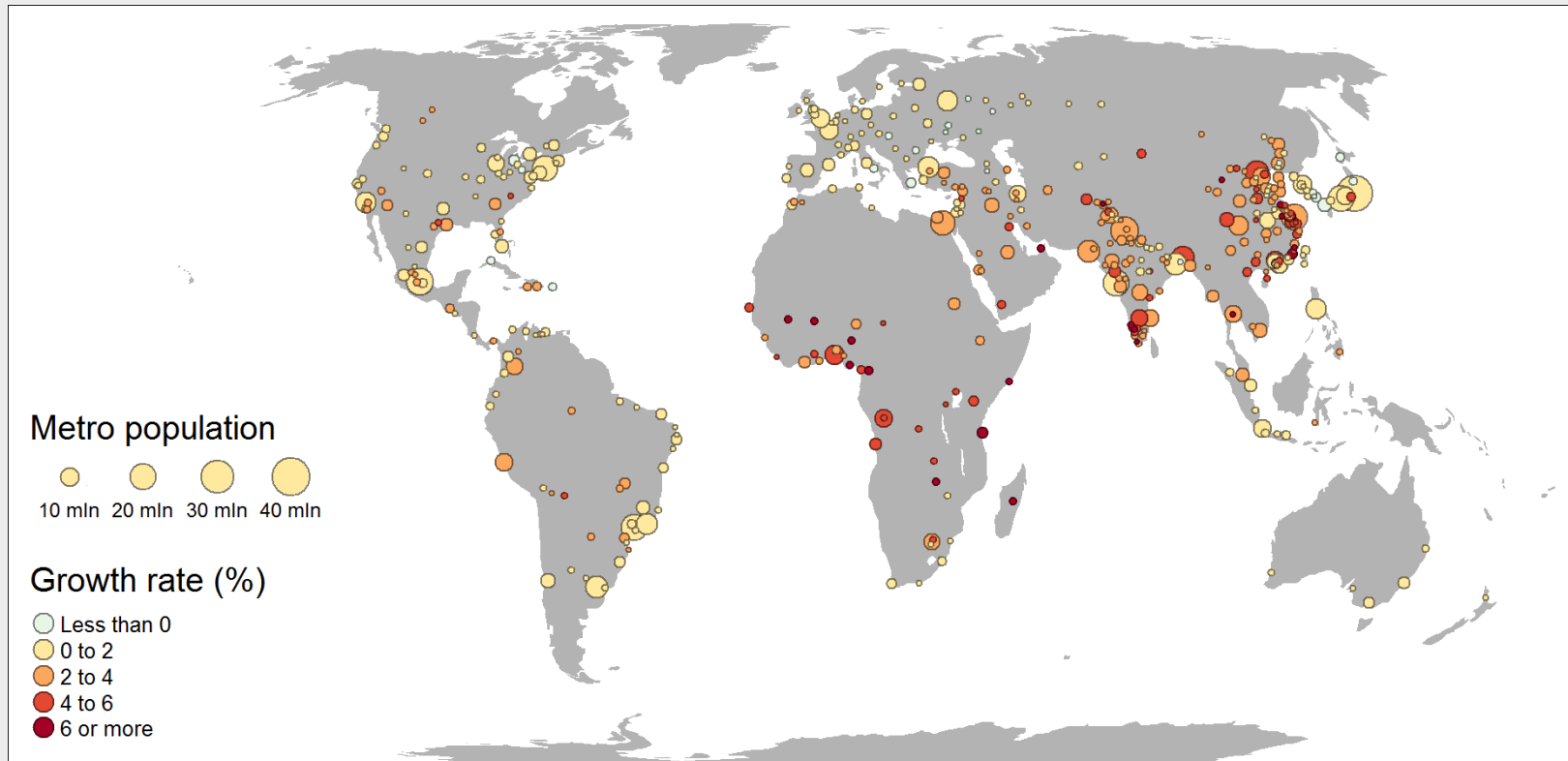
```
tm_shape(world) +  
  tm_polygons("income_grp", palette="-Blues",  
             title="Income classification") +  
  tm_text("iso_a3", size="AREA") +  
  tm_layout_world()
```

RColorBrewer palette  
"Blues" reversed.

Predefined layout settings

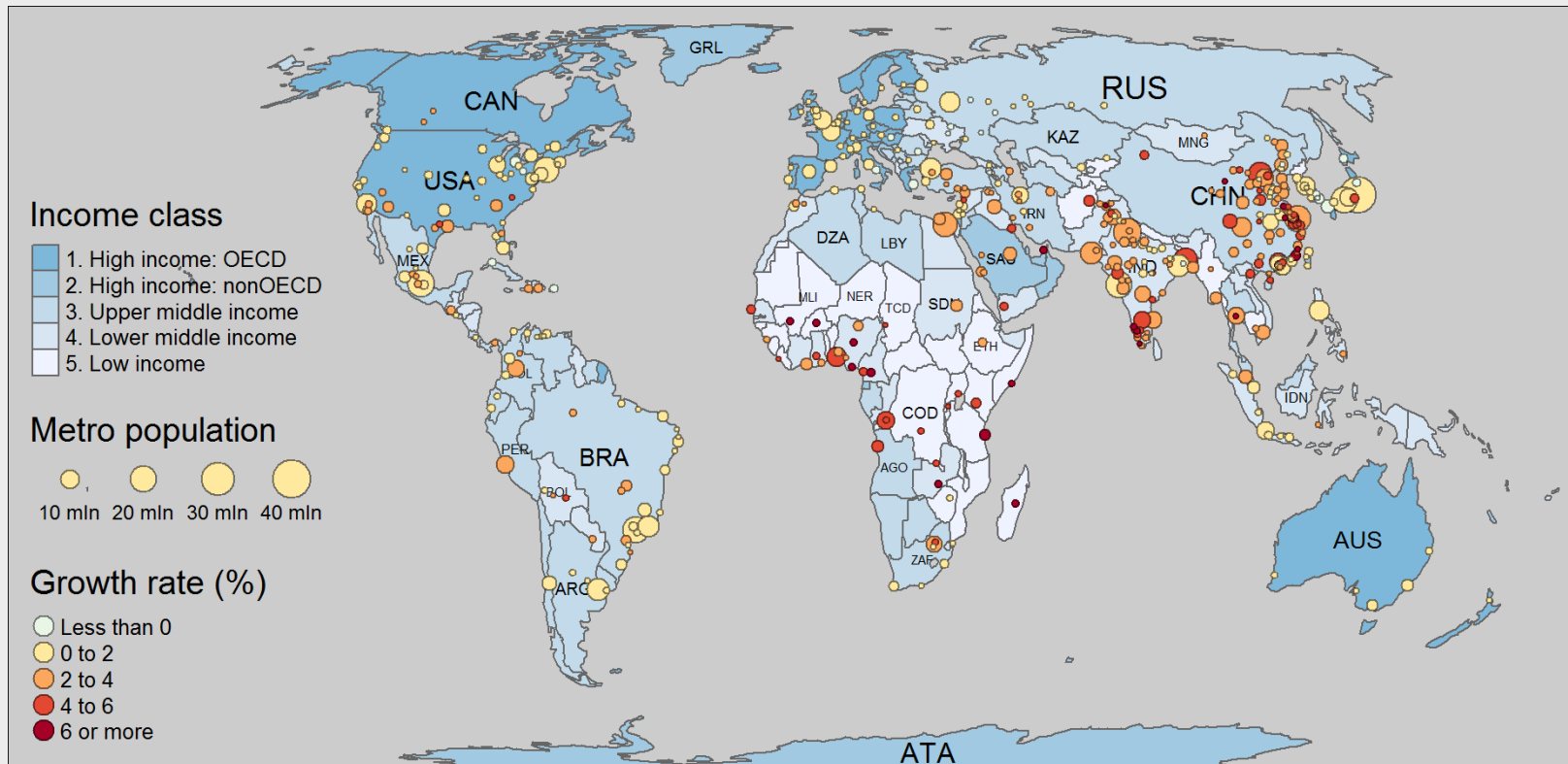


# Example: bubble map



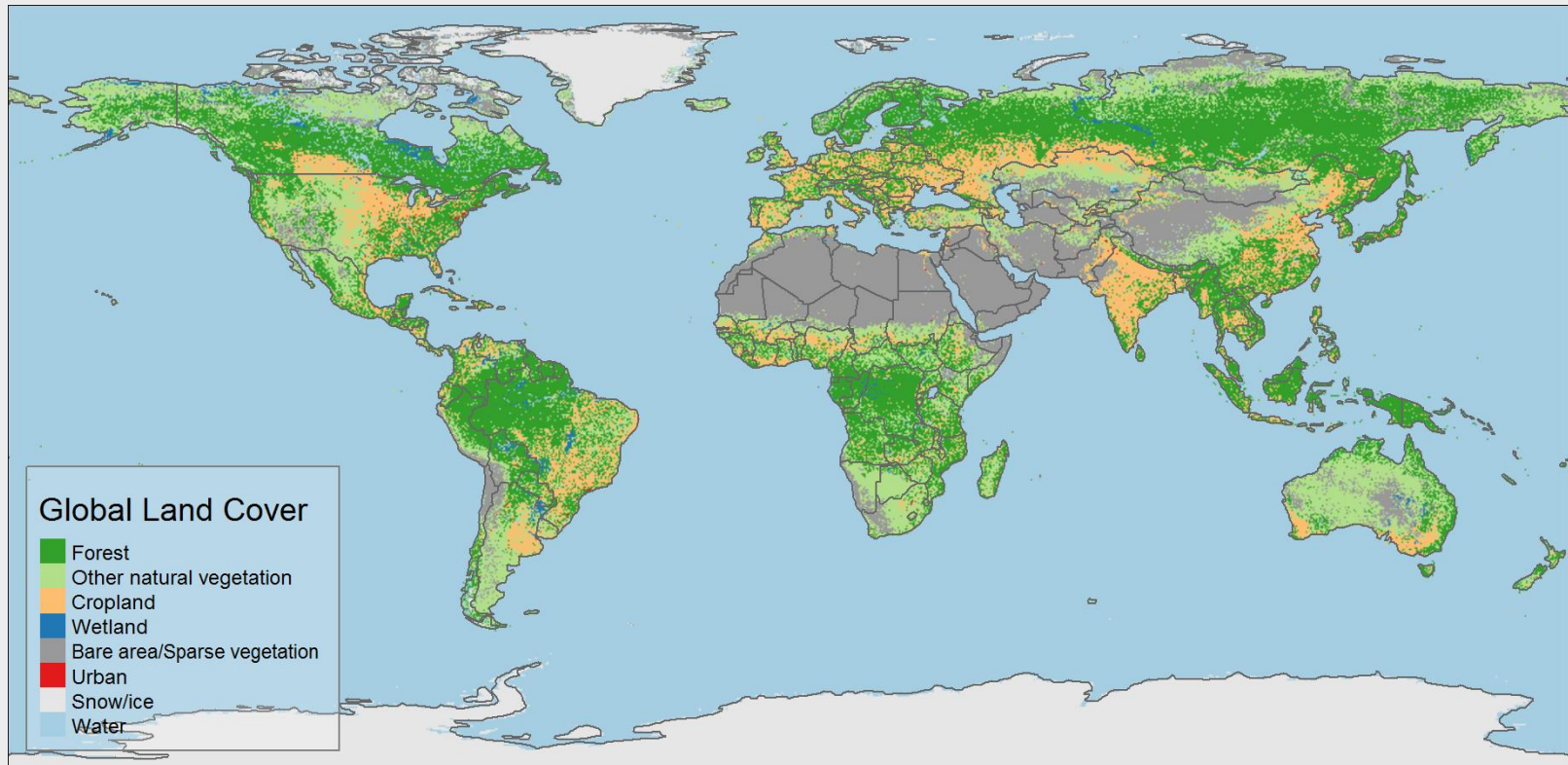
```
tm_shape(world) +  
  tm_fill("grey70") +  
tm_shape(metro) +  
  tm_bubbles("pop2010", col = "growth",  
    border.col = "black", border.alpha = .5, style="fixed",  
    breaks=c(-Inf, 0, 2, 4, 6, Inf), palette="-RdYlBu",  
    title.size="Metro population", title.col="Growth rate (%)") +  
tm_layout_world()
```

# Example: choropleth + bubble map



```
tm_shape(world) +  
  tm_polygons("income_grp", palette="-Blues", contrast = .5,  
    title="Income class",) +  
  tm_text("iso_a3", size="AREA") +  
  tm_shape(metro) +  
  tm_bubbles("pop2010", col = "growth",  
    border.col = "black", border.alpha = .5, style="fixed",  
    breaks=c(-Inf, 0, 2, 4, 6, Inf), palette="-RdYlBu",  
    title.size="Metro population", title.col="Growth rate (%)") +  
  tm_layout_world(bg.color = "gray80")
```

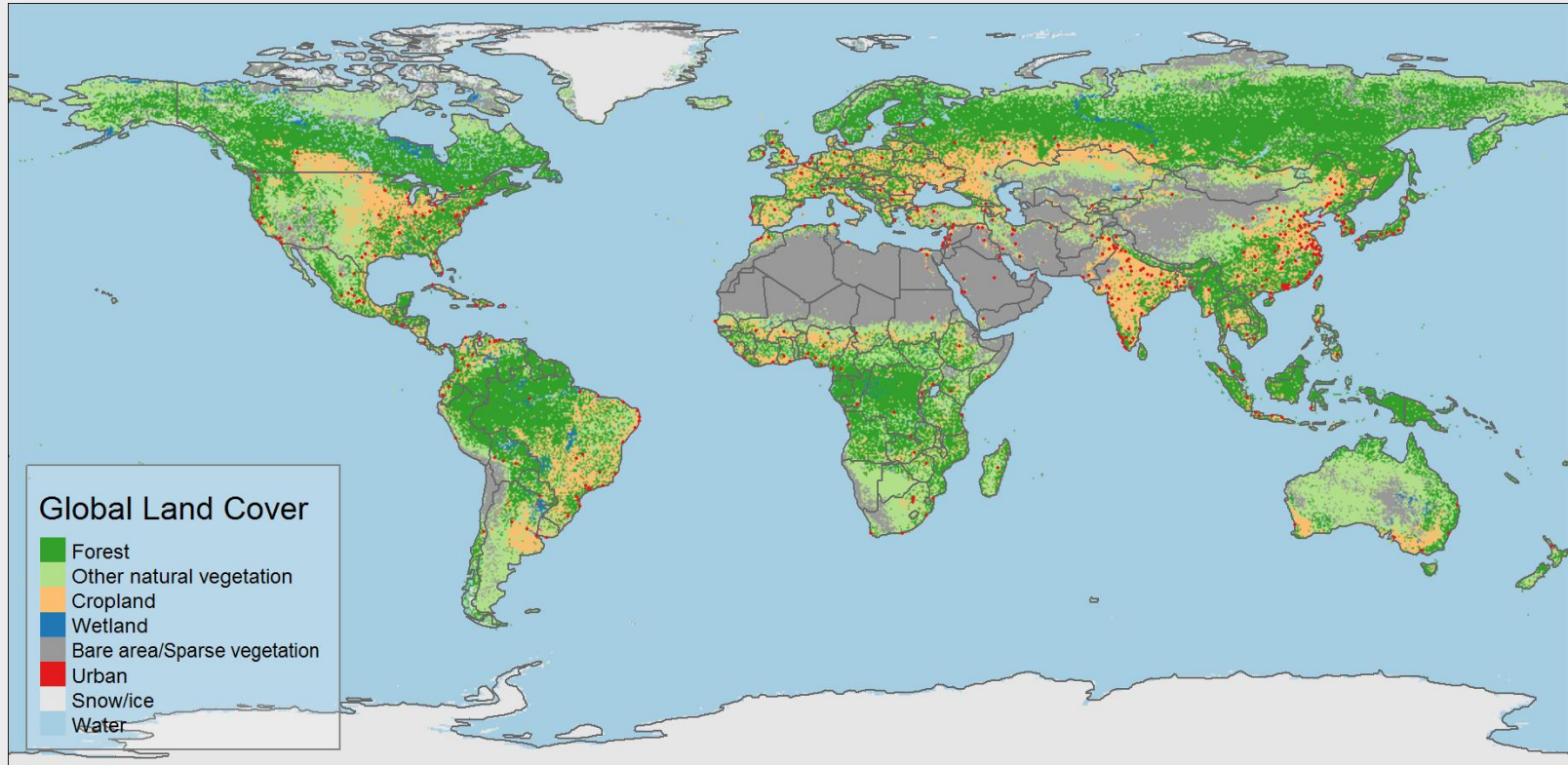
# Example: raster map



```
pa18 <- c("#33A02C", "#B2DF8A", "#FDBF6F", "#1F78B4", "#999999", "#E31A1C", "#E6E6E6", "#A6CEE3")
tm_shape(land, ylim = c(-88,88)) +
  tm_raster("cover_cls", palette = pa18, title="Global Land Cover") +
tm_shape(world) +
  tm_borders() +
tm_layout_world(legend.bg.color = "white", legend.bg.alpha=.2,
                 legend.frame="gray50", legend.width=.2)
```

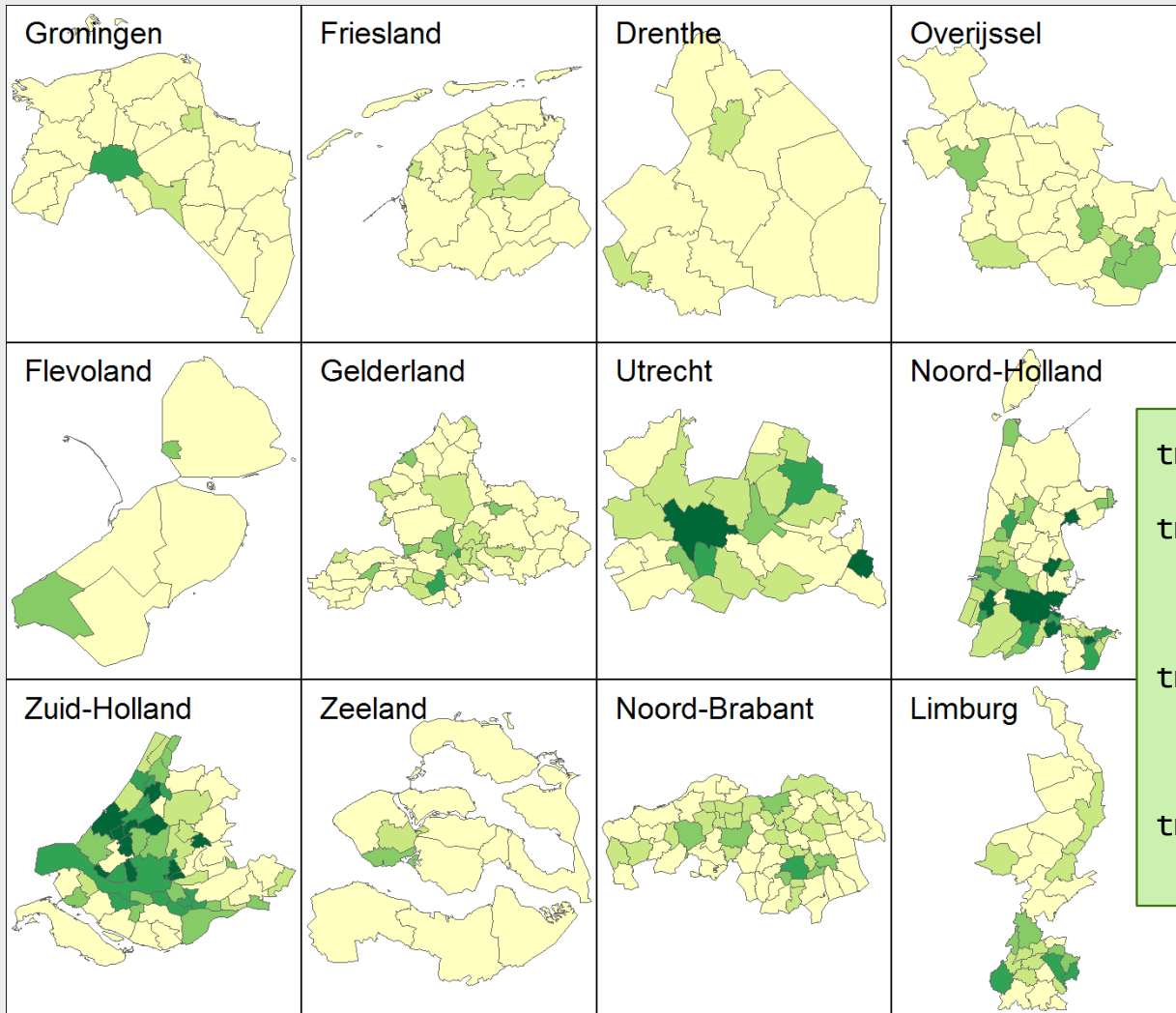


# Example: raster map (with dotmap)



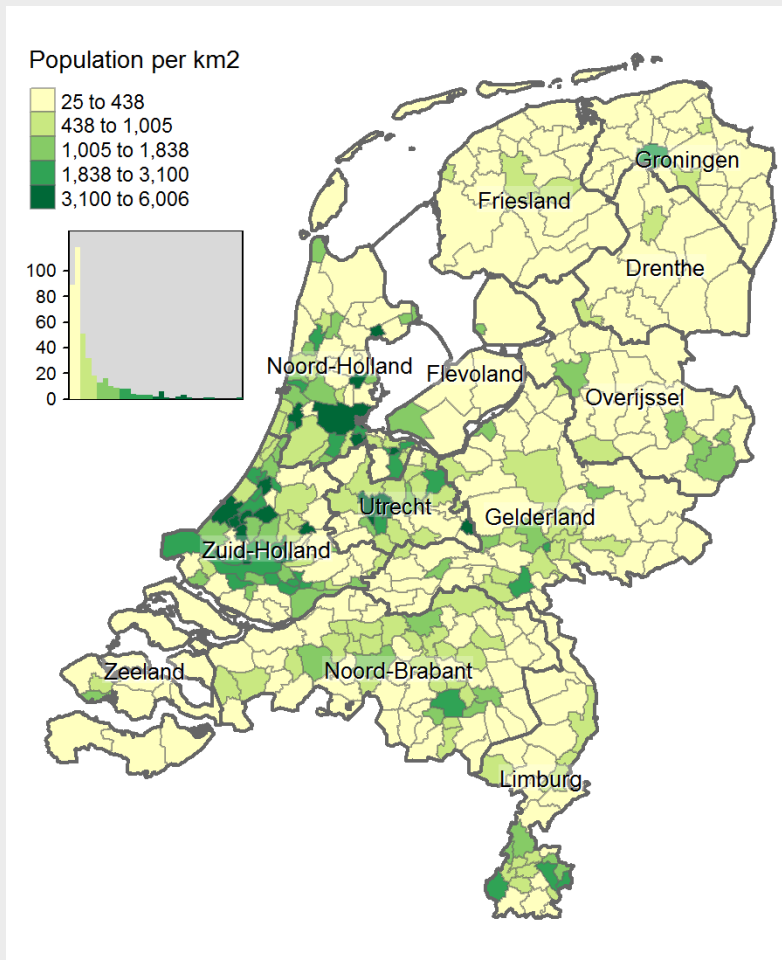
```
pal8 <- c("#33A02C", "#B2DF8A", "#FDBF6F", "#1F78B4", "#999999", "#E31A1C", "#E6E6E6", "#A6CEE3")
tm_shape(land, ylim = c(-88,88)) +
  tm_raster("cover_cls", palette = pal8, title="Global Land Cover") +
tm_shape(world) +
  tm_borders() +
tm_shape(metro) +
  tm_bubbles(size = .01, col = "#E31A1C") +
tm_layout_world(legend.bg.color = "white", legend.bg.alpha=.2,
  legend.frame="gray50", legend.width=.2)
```

# Small multiples



```
tm_shape(NLD_muni) +  
tm_polygons("population",  
style="kmeans",  
convert2density = TRUE) +  
tm_facets(by="province",  
free.coords=TRUE,  
drop.shapes=TRUE) +  
tm_layout(legend.show = FALSE,  
outer.margins=0)
```

# Histogram



```
tm_shape(NLD_muni, projection="rd") +  
  tm_borders(alpha = .5) +  
  tm_fill("population",  
    convert2density = TRUE,  
    style= "kmeans",  
    title="Population per km2",  
    legend.hist = TRUE) +  
tm_shape(NLD_prov) +  
  tm_borders(lwd=2) +  
  tm_text("name", size=0.8,  
    shadow=TRUE, bg.color="white",  
    bg.alpha=.25) +  
tm_layout(draw.frame=FALSE,  
  bg.color="white",  
  inner.margins=c(.02, .05, .02, .02),  
  legend.hist.bg.color = "grey85")
```



# Some convenient functions

Read ESRI shape file: `NLD_muni <- read_shape("NLD_2014_municipality.shp")`

Set map projection: `NLD_muni <- set_projection(NLD_muni, "rd")`

Append data: `NLD_muni <- append_data(NLD_muni, NLD_data,  
key.shp="code", key.data="muni_code")`

Split shapes: `NLD_muni_list <- split(NLD_muni, "name")`

Combine shapes: `NLD_muni2 <- do.call("sbind", NLD_muni_list)`

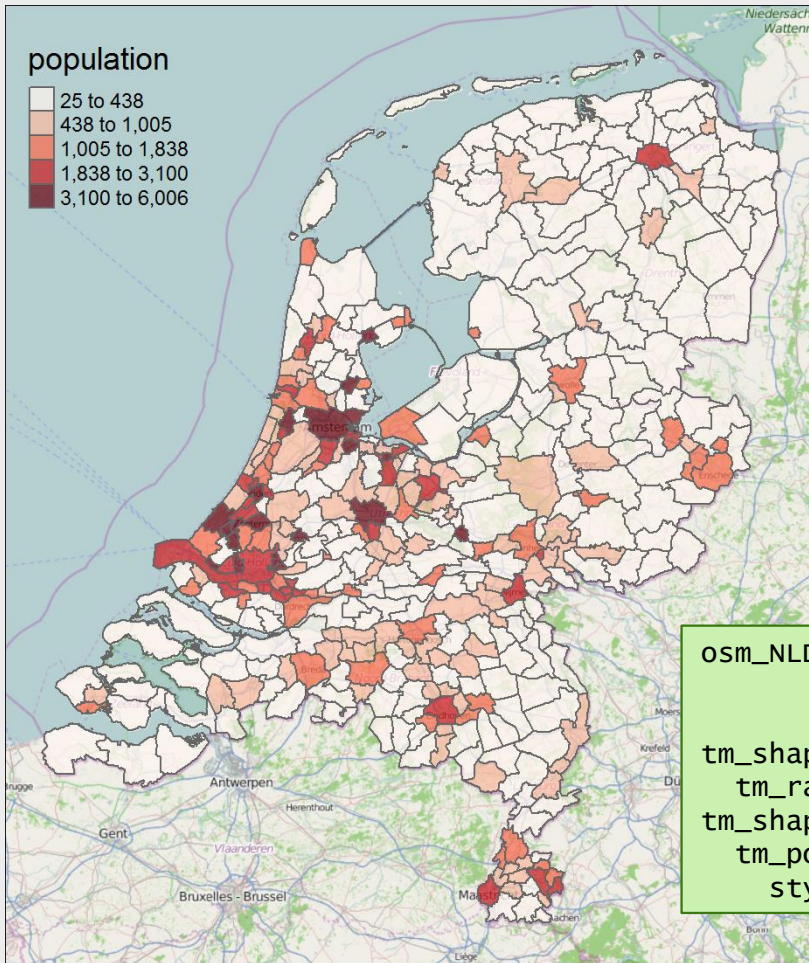
Get aspect ratio: `get_asp_ratio(NLD_muni)`

Create bounding box: `bb(NLD_muni, ext=1.25)  
bb(NLD_muni, projection="longlat")  
bb(q="Aalborg, Denmark")  
...`

New!

# Open Street Map

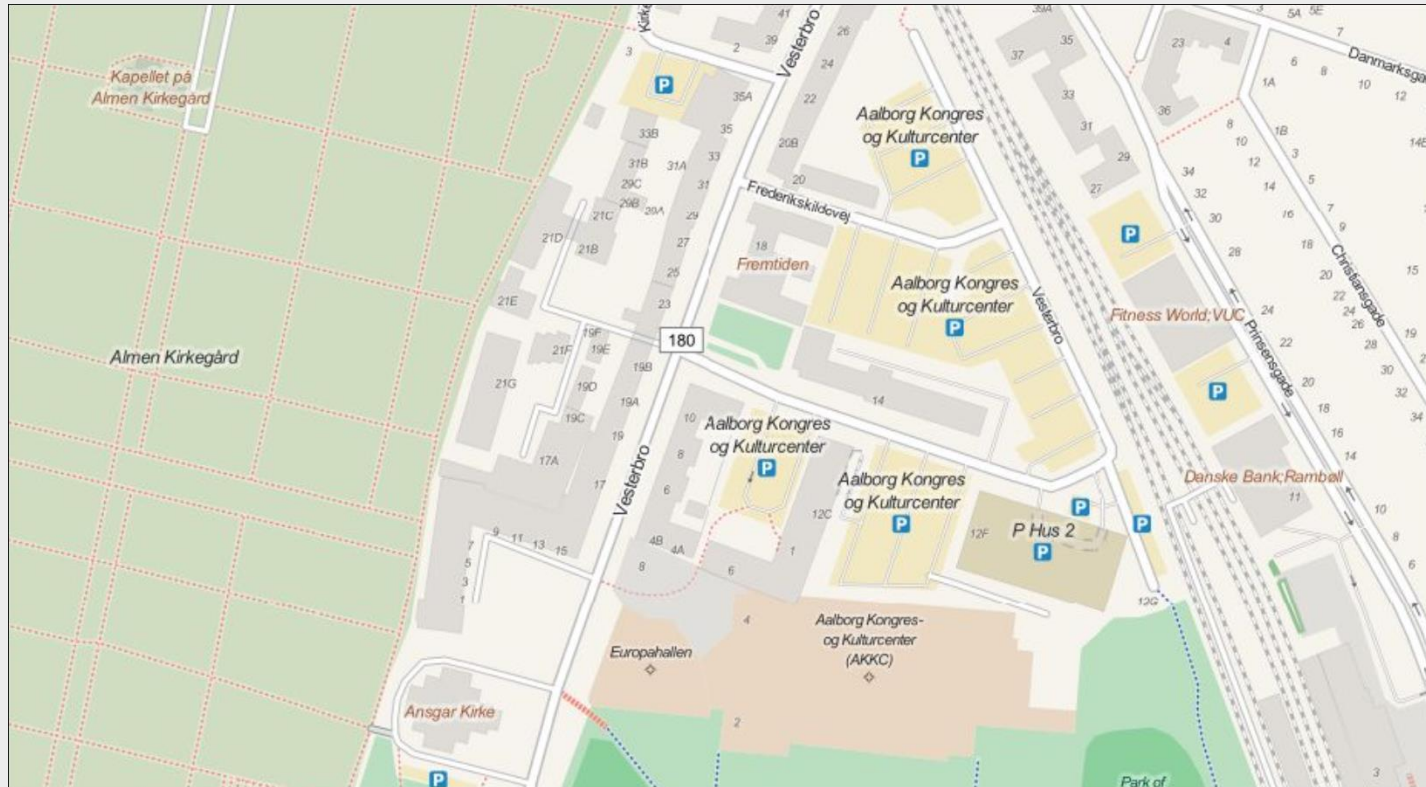
New!



```
osm_NLD <- read_osm(  
  bb(NLD_muni, ext=1.1, projection="longlat")  
  
tm_shape(osm_NLD) +  
  tm_raster() +  
  tm_shape(NLD_muni) +  
  tm_polygons("population", convert2density=TRUE,  
  style="kmeans", alpha=.7, palette="Reds")
```

# Open Street Map

New!



```
# define bounding box:  
bb_Aal <- bb(q="kongres og kulturcenter, Aalborg, Denmark")  
  
# read OSM raster data  
rast_Aal <- read_osm(bb_Aal, type="mapquest")  
  
# plot  
qtm(rast_Aal)
```

# Open Street Map

New!



```
# read OSM vectors
vec_Aa1 <- read_osm(bb_Aa1,
  buildings=osm_poly("building"),
  roads=osm_line("highway"),
  trees=osm_point("natural=tree"),
  park=osm_poly("leisure=park"),
  cemetery=osm_poly("landuse=cemetery"),
  railway=osm_line("railway"),
  parking=osm_poly("amenity=parking"))
```

```
# plot with regular tmap functions
tm_shape(vec_Aa1$park, bbox=bb_Aa1) +
  tm_polygons(col = "darkolivegreen3") +
tm_shape(vec_Aa1$cemetery) +
  tm_polygons(col="darkolivegreen3") +
tm_shape(vec_Aa1$parking) +
  tm_polygons(col="grey85") +
tm_shape(vec_Aa1$building) +
  tm_polygons(col = "gold") +
tm_shape(vec_Aa1$roads) +
  tm_lines("grey40", lwd = 3) +
tm_shape(vec_Aa1$trees) +
  tm_bubbles(size=.25, col="forestgreen") +
tm_shape(vec_Aa1$railway) +
  tm_lines(col = "grey40", lwd = 3, lty =
"longdash") +
tm_layout(inner.margins=0, bg.color="grey95")
```

# Future ideas

- Cartogram
- Flow maps
- Interactive maps (with **htmlwidgets** or **shiny**)

Any other ideas, or suggestions?  
Bugs found?

<https://github.com/mtennekes/tmap>  
Developers are welcome!

