Welcome to

The Session Will Begin Shortly

START

Spatial Data Analysis and Visualization in R

Session 19: Visualization of Raster Data with tmap

instats

Visualization of raster data with tmap

- Class agnostic, i.e. works with **terra** and **stars** objects
- Use `tm_raster()` for raster data layers
- Use `tm_rgb()` for RGB plots

Libraries

```
library(tmap)
library(terra)
library(stars)
library(sf)
library(spDataLarge)
library(cols4all)
```

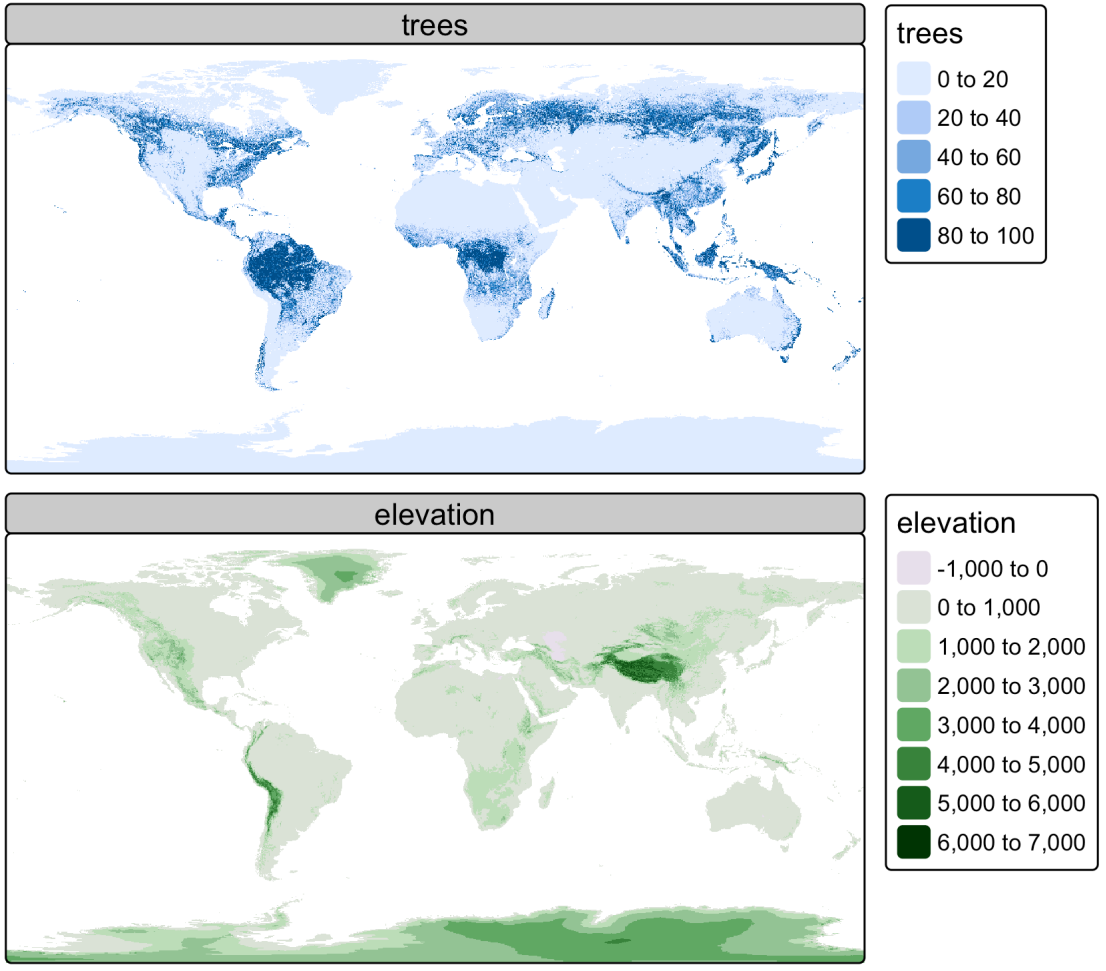
Example data: land from **tmap**

```
land
```

```
stars object with 2 dimensions and 4 attributes
attribute(s):
                cover                                cover_cls
 Water bodies     :393060   Water                        :393060
 Snow / Ice       : 61986   Snow/ice                     : 61986
 Herbaceous       : 21377   Forest                       : 48851
 Tree Open        : 16171   Other natural vegetation     : 32611
 Sparse vegetation: 12247   Bare area/Sparse vegetation: 26904
 Cropland         : 11658   Cropland                     : 17843
 (Other)          : 66701   (Other)                      :  1945
     trees           elevation
 Min.   :  0.00   Min.   :-412
 1st Qu.:  0.00   1st Qu.: 218
 Median :  0.00   Median : 608
 Mean   : 15.59   Mean   :1140
 3rd Qu.: 19.00   3rd Qu.:1941
 Max.   :100.00   Max.   :6410
 NA's   :393060   NA's   :389580
```

Plot trees and elevation

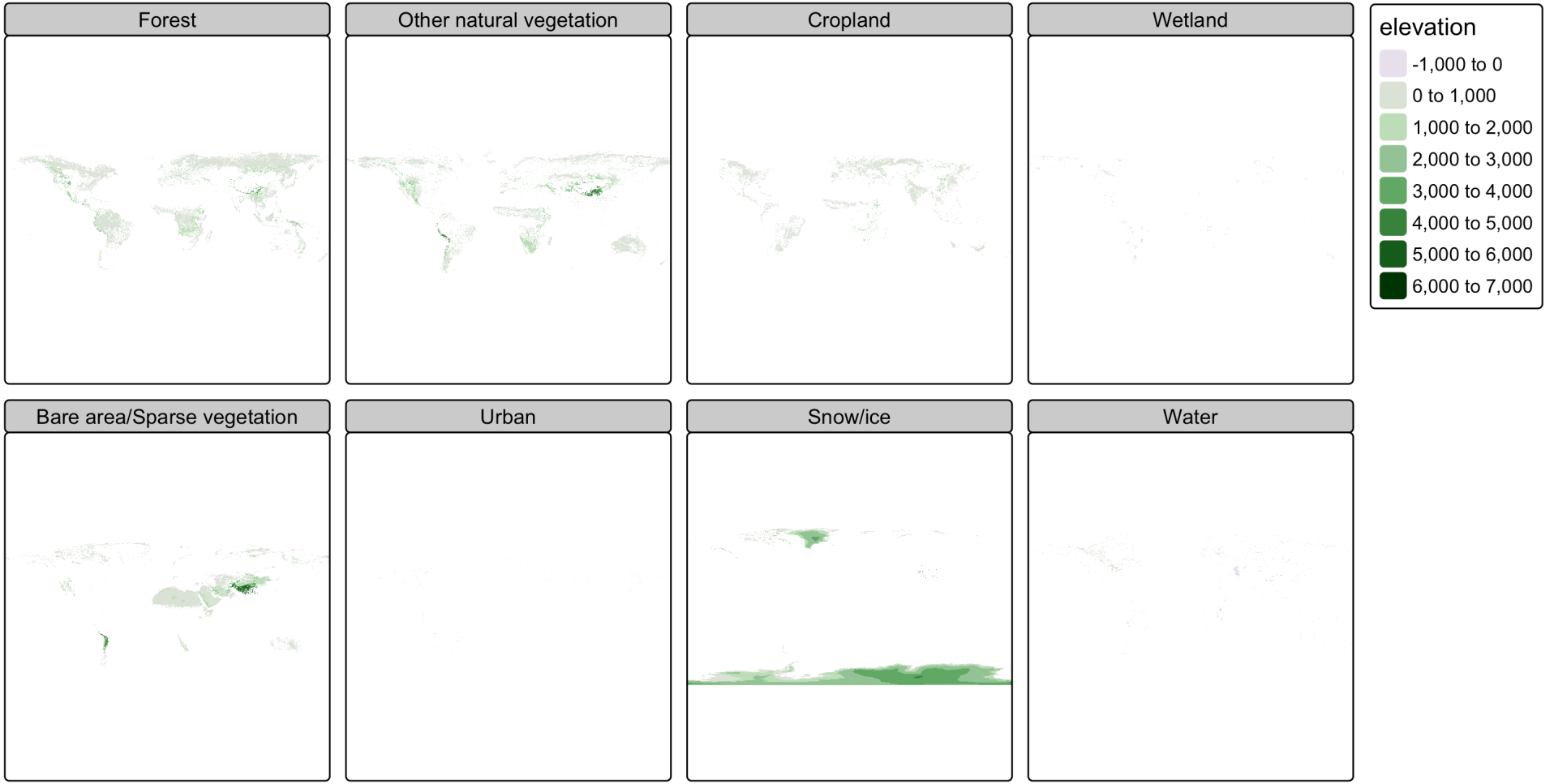
```
tm_shape(land) +
    tm_raster(c("trees", "elevation"))
```



- One facets for each variable
- Different scale is used

Plot elevation for each land cover class

```
tm_shape(land) +
    tm_raster("elevation") +
    tm_facets("cover_cls")
```

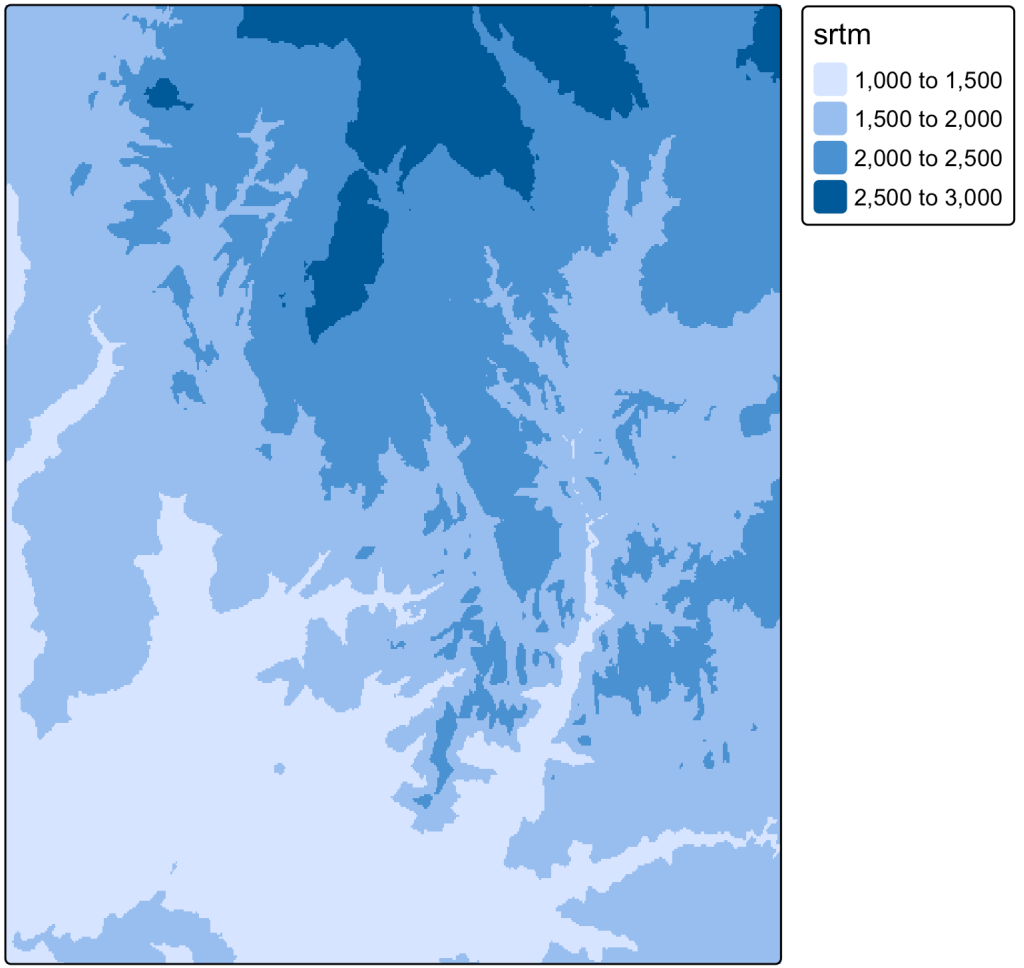


Example data: Zion data from **spDataLarge**

```
srtm = rast(system.file("raster/srtm.tif", package = "spDataLarge"))
zion = read_sf(system.file("vector/zion.gpkg", package = "spDataLarge"))
zion = st_transform(zion, st_crs(srtm))
zion_points <- st_transform(spDataLarge::zion_points, st_crs(srtm))
```

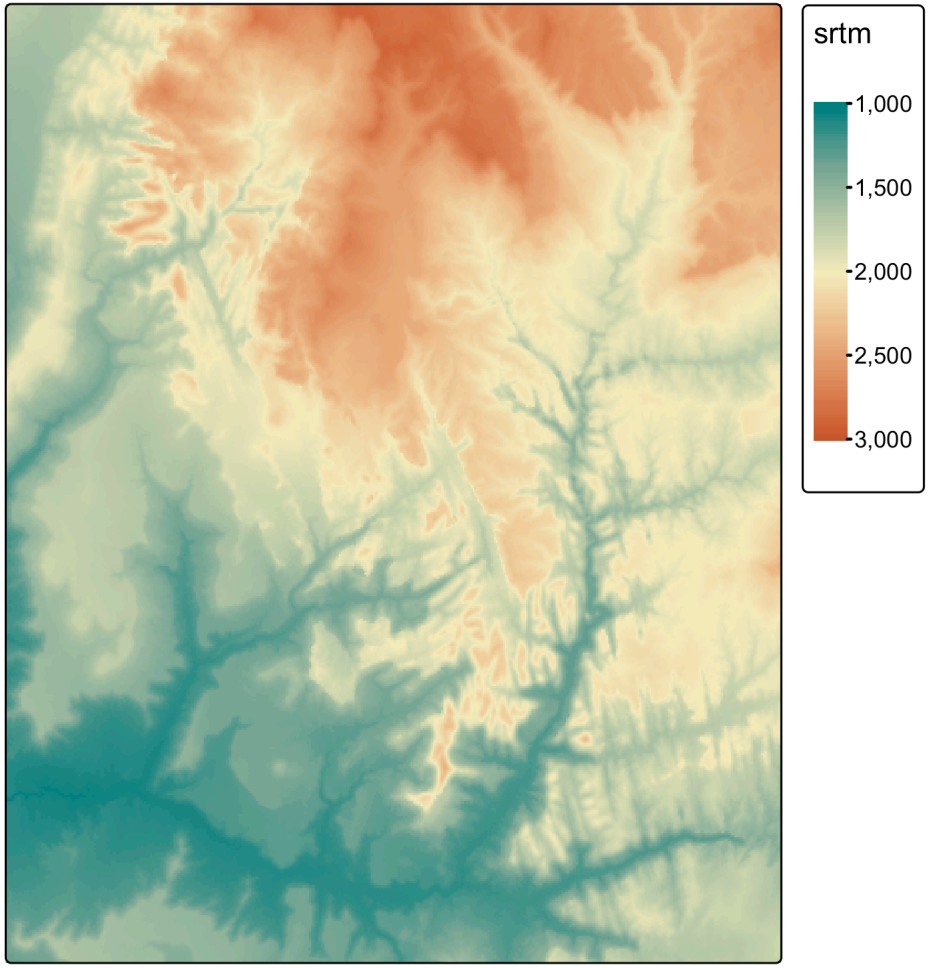
Plot raster

```
tm_shape(srtm) +
    tm_raster()
```



Change scale

```
tm_shape(srtm) +
    tm_raster(col = "srtm",
              col.scale = tm_scale_continuous(values = "carto.geyser", limits = c(1000, 3000)))
```



Example data

```
tif_file <- system.file("tif/L7_ETMs.tif", package = "stars")
L7 <- read_stars(tif_file)
L7
```

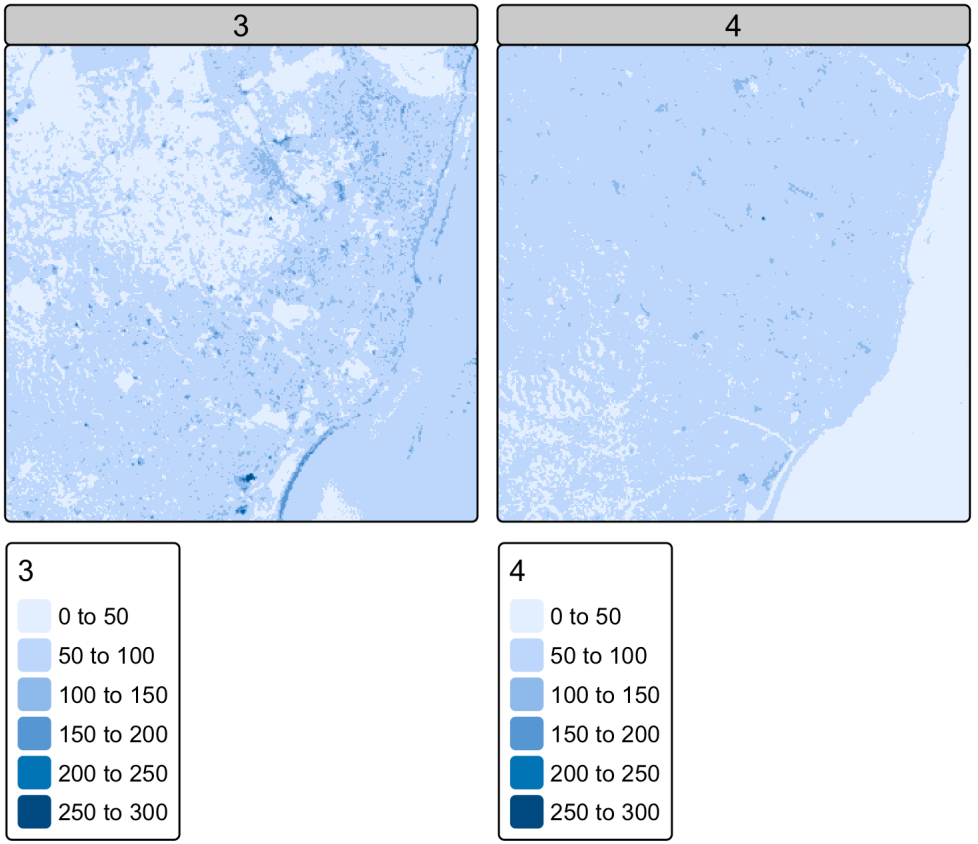
```
stars object with 3 dimensions and 1 attribute
attribute(s):
             Min. 1st Qu. Median     Mean 3rd Qu. Max.
L7_ETMs.tif     1      54     69 68.91242      86  255
dimension(s):
     from  to  offset delta                       refsys point x/y
x       1 349  288776  28.5 SIRGAS 2000 / UTM zone 25S FALSE [x]
y       1 352 9120761 -28.5 SIRGAS 2000 / UTM zone 25S FALSE [y]
band    1   6      NA    NA                           NA    NA
```

Plot each band separately

```
tm_shape(L7) +
    tm_raster()
```

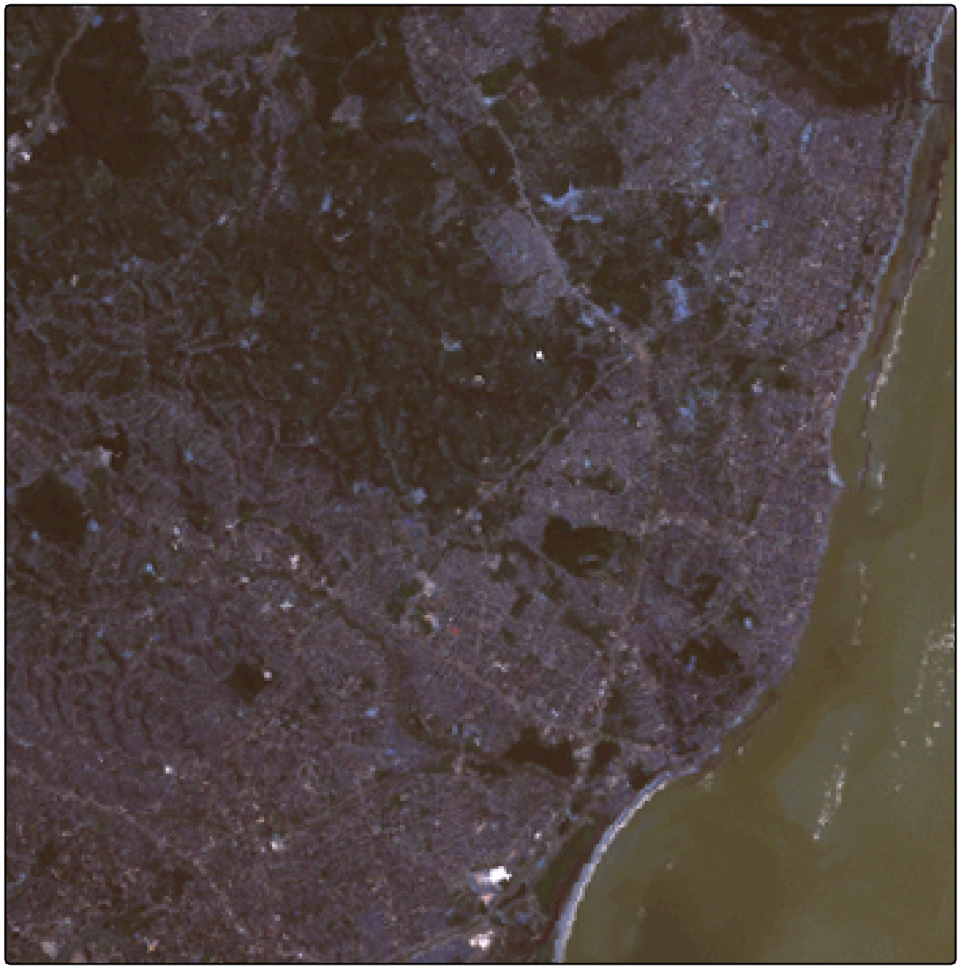
Plot band 3 and 4

```
tm_shape(L7) +
    tm_raster(col = tm_vars(dimvalues = 3:4))
```



Plot RGB using bands 1 to 3

```
tm_shape(L7) +
    tm_rgb(col = tm_vars(dimvalues = 1:3, multivariate = TRUE))
```



Stretch bands using quantiles

```
tm_shape(L7) +
    tm_rgb(col = tm_vars(dimvalues = 1:3, multivariate = TRUE),
           col.scale = tm_scale_rgb(stretch = "percent", probs = c(.02, .98)))
```

Stretch bands using histogram

```
tm_shape(L7) +
    tm_rgb(col = tm_vars(dimvalues = 1:3, multivariate = TRUE),
           col.scale = tm_scale_rgb(stretch = "histogram"))
```

STOP