

# Welcome to **instats**

**The Session Will Begin Shortly**

# START



# Spatial Data Analysis and Visualization in R

Session 20: Exporting Static Maps to Various Formats,  
including bitmap (JPG, PNG) and Vector Formats (SVG,  
PDF)

**instats**

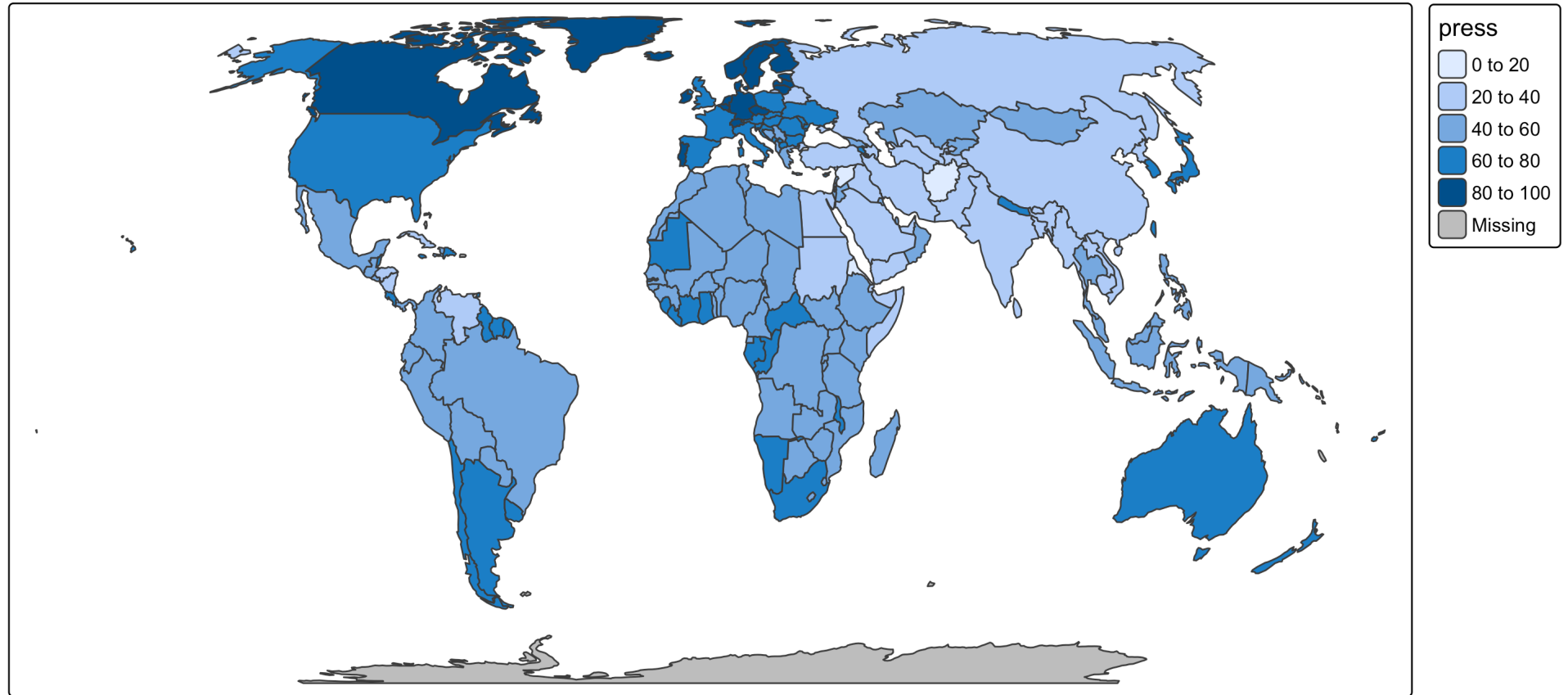
# Content

- Save static plots
- Save interactive plots
- Save animations

# Saving static plots

# Example map

```
library(tmap)
(tm = tm_shape(World) + tm_polygons("press") + tm_crs("auto"))
```



# Save with `tmap_save()`

```
tmap_save(tm, filename = "press_freedom.png", width = 2000, height = 800)
```

- `height` and `width` can be set in pixels or inches
- `dpi` is used to convert them
- Set the width and height for an optimal layout



# Printing devices

- Under the hood, tmap makes use of the core functions from **grDevices**, like `png()` and `pdf()`
- Options from those functions can be passed on, e.g.
  - `type` rendering output type, `"cairo-png"` is the default for png.
  - `colormodel "srgb"` is the default for, but for high print quality, a `"cmyk"` (cyan, magenta, yellow, black) is often required.

# Scaling

Use **scale** to increase or decrease scalable visual elements, like font size and line width

```
tmap_save(tm, filename = "press_freedom_scale_0_5.png", width = 2000, height = 800, scale = 0.5)
```

```
tmap_save(tm, filename = "press_freedom_scale_2_0.png", width = 2000, height = 800, scale = 2)
```

# Saving interactive plots

# Save with `tmap_save()`

And use the html extension

```
tmap_save(tm, filename = "press_freedom.html")
```

# HTML files

- `in.iframe` places the map in an *iframe*, which is a html container.
- `selfcontained` (default **TRUE**) determines if dependencies (JS, CSS etc) are included in the file

```
tmap_save(tm, filename = "press_freedom.html", in.iframe = TRUE)
```

```
tmap_save(tm, filename = "press_freedom.html", selfcontained = FALSE)
```



**STOP**