GIS in R Command Cheat Sheet

*Raster Data*

Nick Eubank

September 25, 2015

---

**Libraries**

- *raster*: tools for raster datasets

---

**Creating Rasters from Scratch**

*RasterLayer (the skeleton):*

```
newRL <- raster(ncol=10, nrow=20, xmn=0, xmx=10, ymn=-10, ymx=10)
```

*RasterLayer w/ Data (skeleton + data):*

```
values(newRL) <- [vector]
```

- Length of vector should match total number of cells in Raster Layer obj
- Vector entries associated with raster cells in order, with top left cell as 1, increasing left to right, then top to bottom, ending with bottom right cell.

---

**Loading Spatial Objects from Files**

```
dem <- raster("file name.fileextension")
```

- Pass the entire filename – path, filename, and extension – unlike in `readOGR()`.

---

**Interrogating Raster and Setting Values**

**Quick summary:** just type name of raster object

**Check if has values:** `hasValues(Raster obj)`

**Viewing or Setting Values:** In general, `raster` commands will return a value if just typed, and will set a value if an assignment is made. So `nrow(Raster obj)` gets number of rows, `nrow(Raster obj)<-5` sets number of rows to 5. Among these:

- **Number of rows, columns, resolution**: `nrow(Raster obj), ncol(Raster obj), res(Raster obj)`
- **Values**: `values(Raster obj)`
Managing Projections

Note: similar to vector data, but without the intermediate CRS() step – just pass the proj4 string.
Assigning projection by EPSG code: `projection([Raster obj]) <- "+init=EPSG:4326"
Get projection from Spatial obj: `projection([Raster obj])`
Re-project:

```
reprojectedRaster <- projectRaster([raster obj], crs=[proj4 string for new projection])
```

• BUT: remember re-projecting rasters is computationally difficulty and can reduce precision, so if you can re-project your Spatial objects instead!

Projection code database