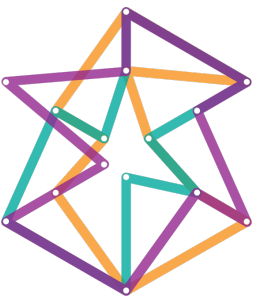


Geospatial Data Carpentry Workshop: : CHEAT SHEET



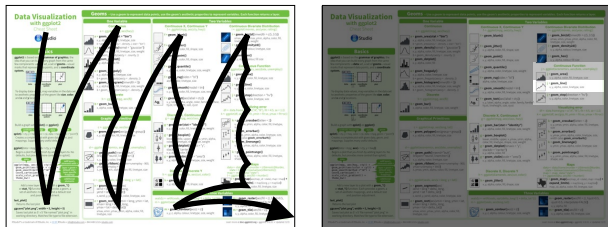
Raster Data

Thank you for making a new cheatsheet for R!
These cheatsheets have an important job:

Cheatsheets make it easy for R users to look up useful information.

Remember that the best cheatsheets are **visual**—not written—documents. Whenever possible use visual elements to make it easier for readers to find the information they need.

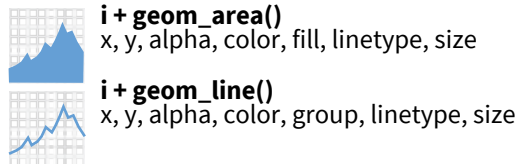
1. Use a **layout** that flows and makes it easy to zero in on specific topics.



2. Use **visualizations** to explain concepts quickly and concisely.

summary function

3. Use visual elements to make the sheet **scannable**.



4. Use visual **emphasis** (like color, size, and font weight) to make important information easy to find.

dplyr::lag() - Offset elements by 1
dplyr::lead() - Offset elements by -1

YOUR LOGO
(optional)

Layout Suggestions

Use headers, colors, and/or backgrounds to **separate or group together sections**.

Section 1

Section 2

Section 3

Create a visual hierarchy. Help users navigate the page with titles, subtitles, and subsubtitles

Title

SUBTITLE

SUBSUBTITLE

Fit sections to content. Try several different layouts.

Use numbers or arrows to link sections if the order/**flow** is confusing.

Quickly identify content with a **package hexsticker** (if available)



Copyright

Each cheatsheet should be licensed under the creative commons license.
To license the sheet as creative commons, put CC'd by <your name> in the small print at the bottom of each page and link it to <http://creativecommons.org/licenses/by/4.0/>

Open Street Map



BOUNDING BOX

With the OSMdata package, it is possible to geocode a spatial text using the Nominatim API. The function `getbb` returns the coordinates of its bounding box: xmin, xmax, ymin and ymax.

```
osmdata::getbb("place name")
```

OVERPASS QUERY

To extract and download Open Street Map (OSM) data into R, we access the Overpass API using a query, to which we add OSM features defined by hierarchical tags called keys and values. To download data about greenhouses for example, the key is "building" and the value "greenhouse".

```
osmdata::opq(bbox) |>  
  add_osm_feature(key, value) |>  
  osmdata_sf()
```

Format of resulting object (sf object)

The result of this query can contain **points**, **lines** and/or **polygons**, each described by a data frame.

Interactive mapping

The **leaflet** package provides a way to create map with interactive features such as zoom, popups, image overlay, etc.



```
leaflet(x) |>  
  addTiles() |>  
  addPolygons()
```

Background map

Added geometries from x

GIS functions

BUFFER

A buffer corresponds to a circular polygon around an 'x' feature with a specified distance 'dist'

```
sf::st_buffer(x, dist)
```



UNION

A union corresponds to the combination of polygons by removing internal boundaries

```
sf::st_union(x, y, ...) |>  
  sf::st_cast(to = "POLYGON") |>  
  sf::st_as_sf()
```

Format of resulting object

Type of resulting object

CENTROID

A centroid corresponds to the centre of mass of a geometric object.

Disables geographic projection

```
sf::sf_use_s2(FALSE) |>  
  sf::st_centroid(x) |>  
  sf::st_transform(., crs)
```

Reproject the resulting object

INTERSECT / INTERSECTION

'Intersect' tests whether geometric objects x and y intersect each other. 'Intersection' performs the intersection and returns an object of the same type as x.

```
sf::st_intersection(x, y)
```

AREA

Computation of the area of a set of features x

Specifies area unit

```
sf::st_area(x) |>  
  units::set_units(., km^2)
```