



Grupo de Investigación da USC

# VISUALIZACIÓN DE LA DIRECCIÓN FLUVIAL.

Dr. Dominic  
ROYÉ

Departamento de Geografía



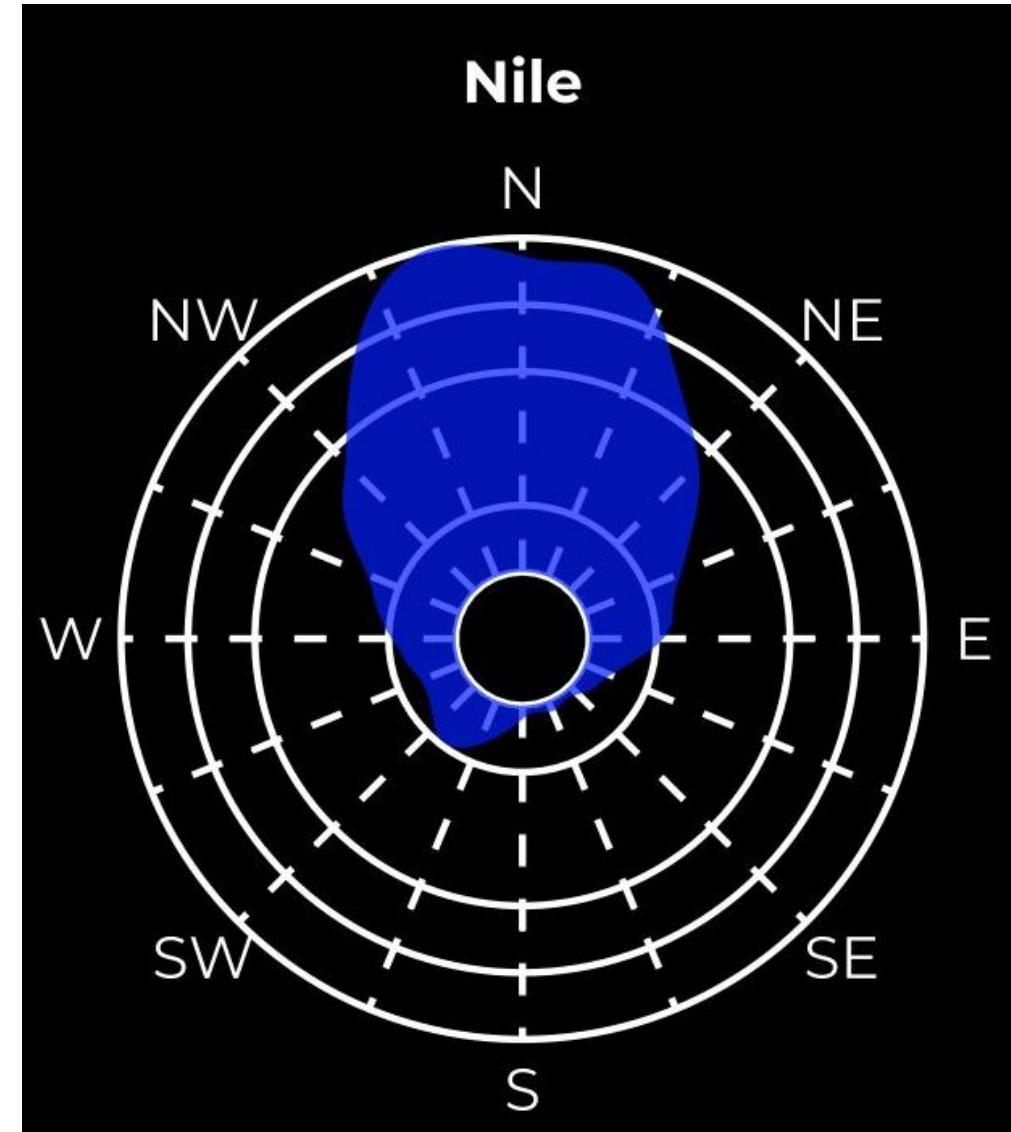
[dominicroye.github.io](http://dominicroye.github.io)



[dominic.roye@usc.es](mailto:dominic.roye@usc.es)

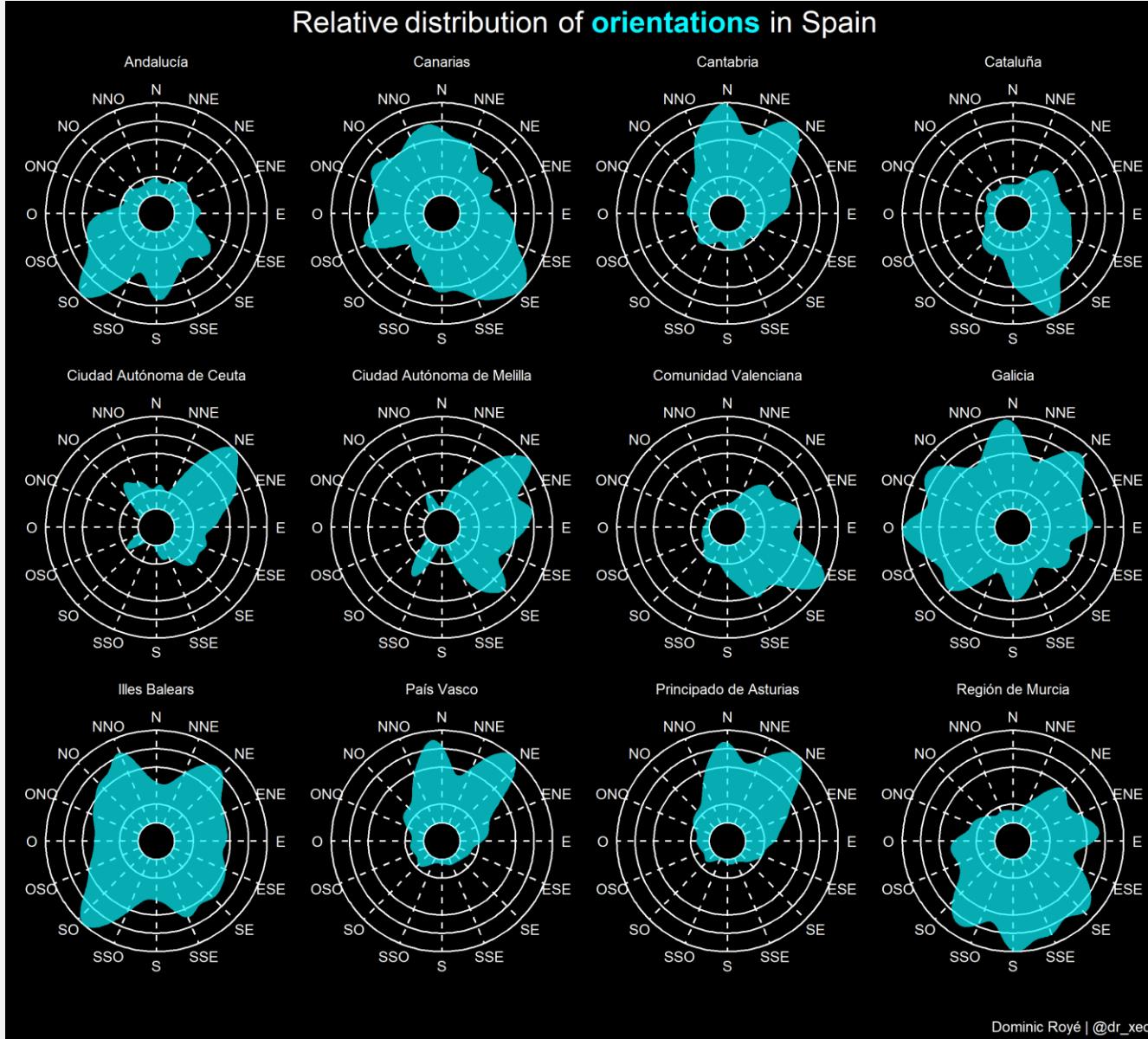


@dr\_xeo



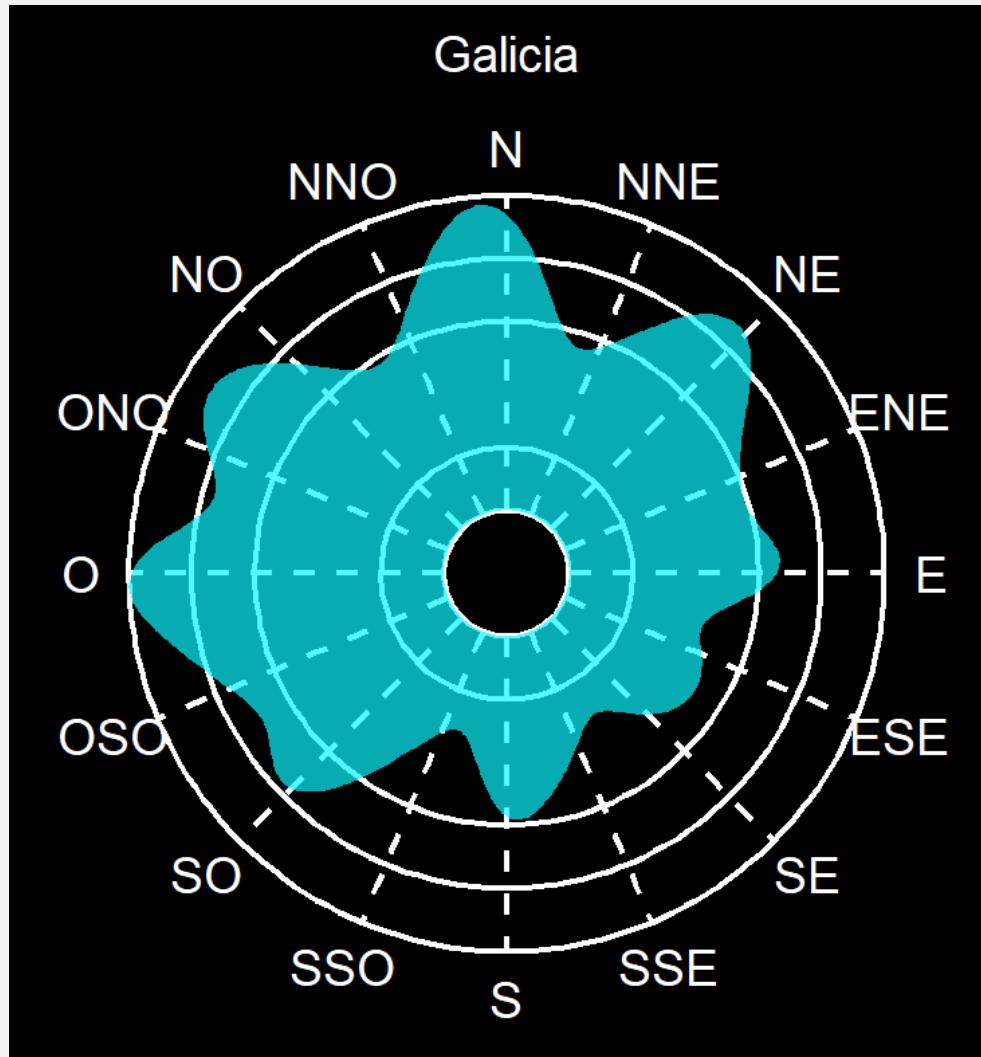
# CONTEXTO

¿Cómo se distribuyen las orientaciones de las costas o la dirección fluvial?



# CONTEXTO

¿Cómo se distribuyen las orientaciones de las costas o la dirección fluvial?



# CONTEXTO

El número de vértices depende de la complejidad y en consecuencia de la resolución.

Río Danubio

*origen*

*desembocadura*

Zona con cambio tectónico

vértices

fractalidad

# CÁLCULO DE LOS ÁNGULOS

*Paquetes*

**library**(geosphere)

## Opción 1

Por ejemplo, imaginémonos dos puntos, Madrid (-3.71, 40.43) y Barcelona (2.14, 41.4). ¿Cuál es el ángulo de su línea recta?

```
bearingRhumb(c(-3.71, 40.43), c(2.14, 41.4))
```

```
## [1] 77.62391
```

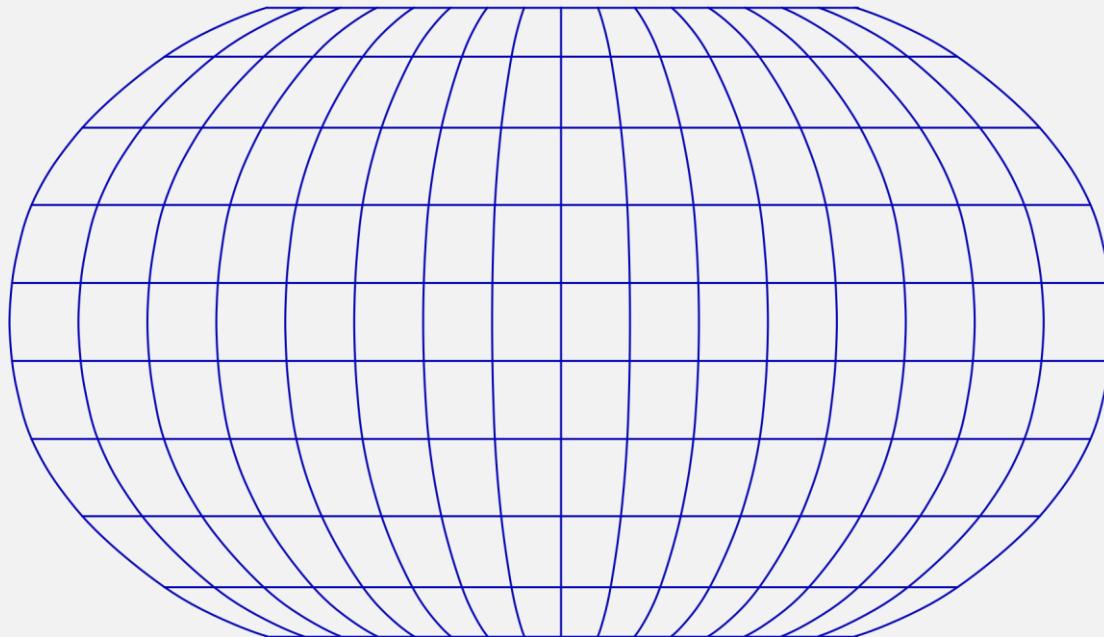
Problema 1: Los vértices deben ir en orden geográfico.

Problema 2: Debemos conocer en qué dirección nos movemos. Madrid a Barcelona ( $77^\circ$ ) o Barcelona a Madrid ( $257^\circ$ ).

# CÁLCULO DE LOS ÁNGULOS

## Opción 2

Usar los atributos de los sistemas de coordenadas proyectados (proyección Robinson, etc) que incluyen el ángulo entre los vértices.



# IMPORTAR Y PROYECTAR

Paquetes  
library(sf)

```
proj_rob <- "+proj=robin +lon_0=0 +x_0=0 +y_0=0 +ellps=WGS84 +datum=WGS84  
+units=m no_defs"  
  
river_line <- st_read("RiverHRCenterlinesCombo.shp") %>%  
  st_zm() %>%  
  st_transform(proj_rob)
```

```
## Reading layer `RiverHRCenterlinesCombo' from data source  
'C:\Users\xeo19\Documents\GitHub\blogR_update\content\post\es\2020-07-24-  
direcciones-del-flujo-fluvial\RiverHRCenterlinesCombo.shp' using driver 'ESRI Shapefile'  
## Simple feature collection with 78 features and 6 fields  
## geometry type: MULTILINESTRING  
## dimension: XYZ  
## bbox: xmin: -164.7059 ymin: -36.97094 xmax: 151.5931 ymax: 72.64474  
## z_range: zmin: 0 zmax: 0  
## geographic CRS: WGS 84
```

# EXTRAER VÉRTICES

*Paquetes*

**library(sf)**

**library(RQGIS3)**

```
# rutas a QGIS
set_env()
# inicio de QGIS Python
open_app()

river_vertices <- run_qgis(alg = "native:extractvertices",
                           INPUT = river_line,
                           OUTPUT = file.path(tempdir(),
                                              "rivers_world_vertices.geojson"),
                           load_output = TRUE)
```

*Requisito de RQGIS3: instalar OSGeo4W*

# ESTIMACIÓN DE LA DENSIDAD

*Paquetes*  
**library**(circular)

Función personalizada

```
dens_circ <- function(x){  
  
  dens <- density.circular(circular(x$angle, units = "degrees"),  
                           bw = 70, kernel = "vonmises",  
                           control.circular = list(units = "degrees"))  
  
  df <- data.frame(x = dens$x, y = dens$y/max(dens$y))  
  
  return(df)  
}
```

Optimización para la banda con **bw.nrd.circular()**

# ESTIMACIÓN DE LA DENSIDAD

*Paquetes*

**library**(circular)

**library**(tidyverse)

Mapear la función sobre cada río

```
dens_river <- split(river_vertices, river_vertices$NAME) %>%
  map_df(dens_circ, .id = "river")
# resultado
head(dens_river)
```

```
##      river      x      y
## 1 Amazon 0.000000 0.2399907
## 2 Amazon 0.704501 0.2492548
```

```
library(showtext)
library(sysfonts)
library(ggtext)
library(tidyverse)
```

# VISUALIZACIÓN

Añadimos una nueva fuente

```
# descarga de fuente
font_add_google("Montserrat", "Montserrat")
# usar showtext para fuentes
showtext_opts(dpi = 200) showtext_auto()
```

Creamos una cuadrícula personalizada

```
grid_x <- tibble(x = seq(0, 360 - 22.5, by = 22.5),
                  y = rep(0, 16),
                  xend = seq(0, 360 - 22.5, by = 22.5),
                  yend = rep(Inf, 16))
```

# VISUALIZACIÓN

*Paquetes*

**library**(showtext)  
**library**(sysfonts)  
**library**(ggtext)  
**library**(tidyverse)

Definimos el título con html y pie de figura

```
# título con html  
title <- "Relative distribution of river <span style='color:#011FFD;'><strong>flow  
direction</strong></span> in the world"  
  
caption <- "Based on data from Zeenatul Basher, 20180215"
```

La función **element\_textbox()** en **theme()** del paquete **ggtext** permite usar texto enriquecido con html.

# VISUALIZACIÓN

Definimos los estilos globales

```
theme_polar <- theme_minimal() +  
  theme(axis.title.y = element_blank(),  
        axis.text.y = element_blank(),  
        legend.title = element_blank(),  
        plot.title = element_textbox(family = "Montserrat",  
                                     hjust = 0.5, colour = "white", size = 15),  
        plot.caption = element_text(family = "Montserrat", colour = "white"),  
        axis.text.x = element_text(family = "Montserrat", colour = "white"),  
        strip.text = element_text(family = "Montserrat",  
                                 colour = "white", face = "bold"),  
        panel.background = element_rect(fill = "black"),  
        plot.background = element_rect(fill = "black"),  
        panel.grid = element_blank()  
)
```

library(showtext)  
library(sysfonts)  
library(ggtext)  
library(tidyverse)

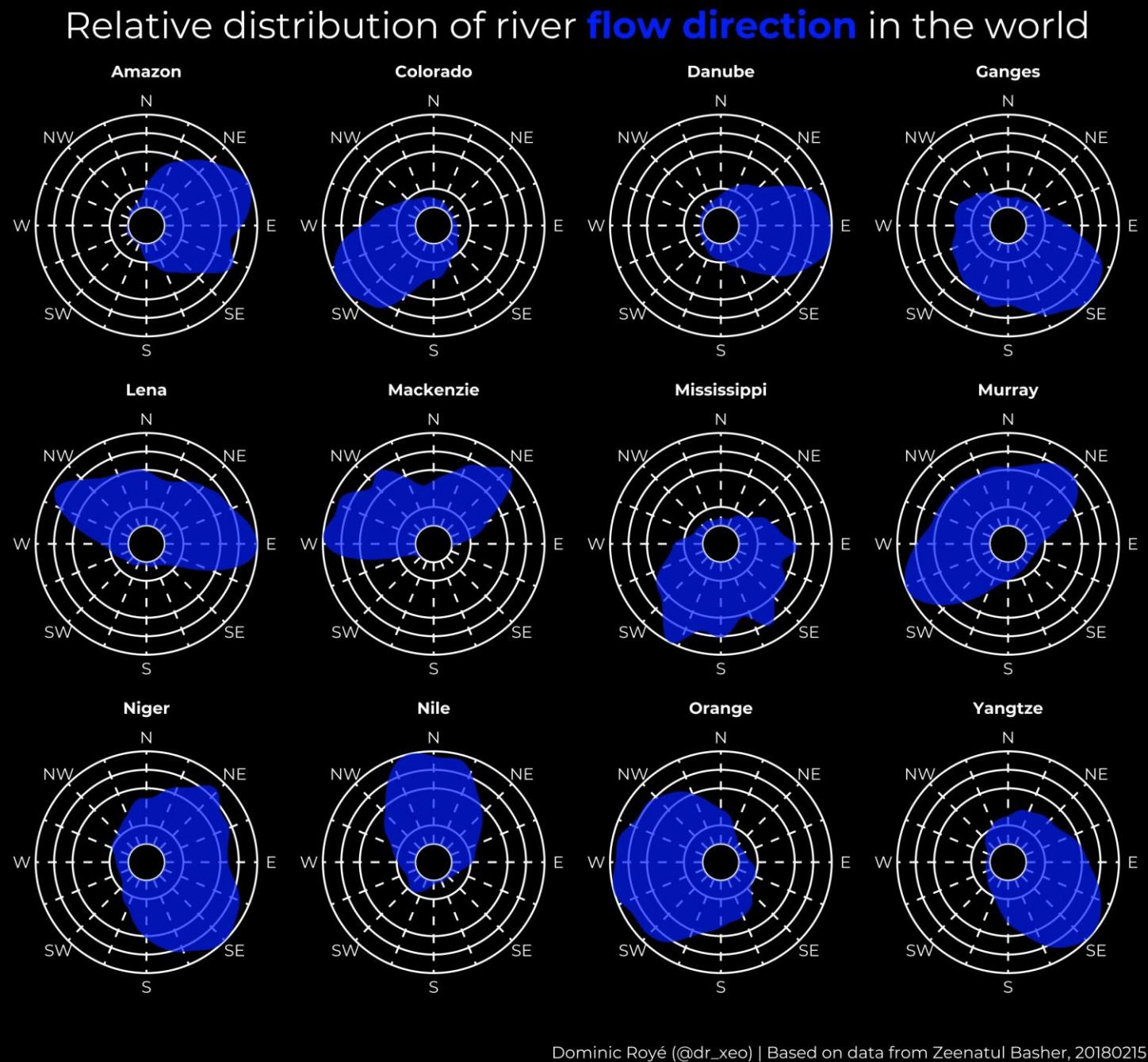
```
library(showtext)
library(sysfonts)
library(ggtext)
library(tidyverse)
```

# VISUALIZACIÓN

Construcción final del gráfico

```
ggplot() + geom_hline(yintercept = c(0, .2, .6, .8, 1), colour = "white") +
  geom_segment(data = grid_x,
    aes(x = x, y = y, xend = xend, yend = yend),
    linetype = "dashed", col = "white") +
  geom_area(data = dens_river,
    aes(x = x, y = y, ymin = 0, ymax = y),
    alpha = .7,
    colour = NA,
    show.legend = FALSE, fill = "#011FFD") +
  scale_y_continuous(limits = c(-.2, 1), expand = c(0, 0)) +
  scale_x_continuous(limits = c(0, 360),
    breaks = seq(0, 360 - 22.5, by = 22.5),
    minor_breaks = NULL,
    labels = c("N", "", "NE", "", "E", "", "SE", "",
              "S", "", "SW", "", "W", "", "NW", ""))
  coord_polar() +
  facet_wrap(river ~ ., ncol = 4) + labs(title = title, caption = caption, x = "") +
  theme_polar
```

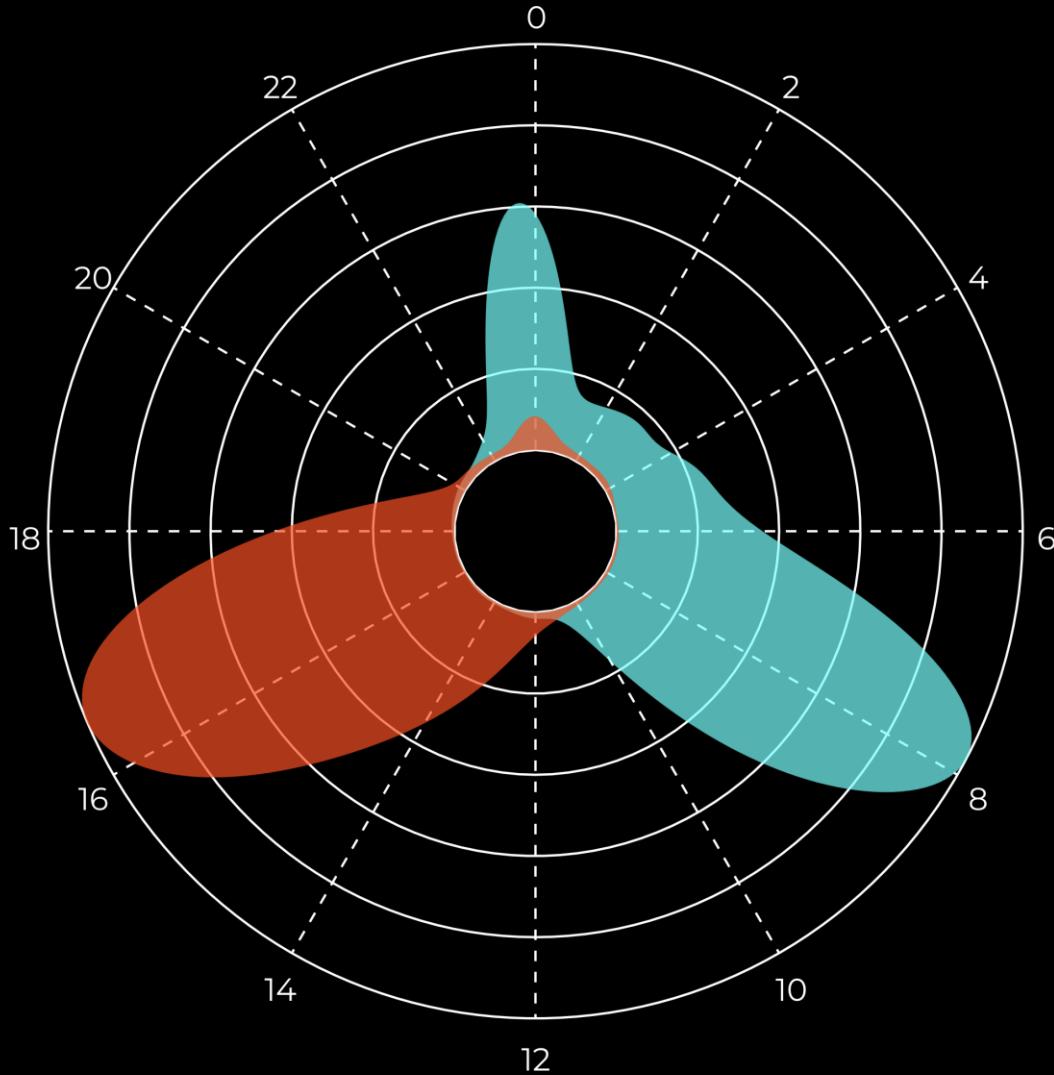
# VISUALIZACIÓN



# MÁS

Distribución de la hora a la que se alcanza la temperatura **máxima** y la **mínima** en Santiago

2006-2019



Dominic Royé (@dr\_xeo) | Datos: Meteogalicia