

On the analysis of the Jenkinson-Collison classification suitability worldwide

Paper notebook - submitted to Geoscientific Model Development

J. A. Fernández-Granja, S. Brands, A. Casanueva, J. Bedia & J. Fernández

2022-01-12

Abstract

This is an example notebook illustrating some primary calculations undertaken in the paper. It is not intended to provide full reproducibility of the results, but a sample on how to achieve this using the [climate4R framework](#), used in the paper. To this aim, we provide two exercises calculating two different types of results. For these examples, we use public datasets only, namely the NCEP-NCAR Reanalysis1 (NCEP), considering historical simulations.

Contents

1	Used packages	2
1.1	Cloud computing with the climate4R Hub	2
2	Analysis of Jenkinson-Collison Weather Types (JC-WT) from NCEP reanalysis in the South Hemisphere and method suitability	3
2.1	Loading reanalysis data	4
3	Assessment of JC-WTs frequencies profiles for the different IPCC regions of the globe.	10
3.1	Loading JC-WTs reanalysis products from Zenodo:	11
3.2	Post-processing with <code>geoprocessor</code> :	11
3.3	Frequency profiles from the grid-boxes of the IPCC region: . . .	13
4	References	15
5	Session info	17

1 Used packages

To ensure the reproducibility of the paper results as accurately as possible, it is recommended to install the package versions used to compile this notebook. The appropriate package versions are indicated here through their version tags using the `devtools` package function `install_github` (Wickham *et al.* 2020), or alternatively, their commit has:

```
devtools::install_github(c("SantanderMetGroup/loader.java@v1.1.1",  
                           "SantanderMetGroup/climate4R.UDG@0.1.1",  
                           "SantanderMetGroup/loader@v1.6.1",  
                           "SantanderMetGroup/transformer@v2.1.3",  
                           "SantanderMetGroup/visualizeR@v1.6.0",  
                           "SantanderMetGroup/geoprocessor@v0.2.0"))
```

The `rgdal` (Bivand, R. *et al.* 2015) package is required for the `geoprocessor` package installation:

```
install.packages("rgdal")
```

Alternatively, and updated image of the packages can be installed using the [conda recipe for climate4R](#).

1.1 Cloud computing with the climate4R Hub

Furthermore, there is a [docker climate4R](#) installation available. The docker file also includes the [jupyter](#) framework enabling a direct usage of `climate4R` via the **climate4R Hub**, a cloud-based computing facility to run `climate4R` notebooks on the cloud using the [IFCA/CSIC Cloud](#)

[Services](#)).

The `climate4R` packages used in this experiment are next loaded:

```
require(loader)
require(transformer)
require(visualizeR)
require(rgdal)
require(geoprocessorR)
```

Additional packages will be used for convenience. For instance, the package `magrittr` (Bache and Wickham 2014) allows to conveniently concatenate functions via the pipe operator `%>%`. In addition, `sp` (Pebesma and Bivand 2005, Bivand *et al.* 2013) is used for plotting.

```
require(magrittr)
require(sp)
```

2 Analysis of Jenkinson-Collison Weather Types (JC-WT) from NCEP reanalysis in the South Hemisphere and method suitability

In this first example, the Jenkinson and Collison Weather Types (JC-WT, 1977) will be calculated in two different locations, and then, the method suitability will be discussed for these two locations in terms of the number of WTs observed and the Unclassified (U) type relative frequency (in a similar way to Fig. 3 from the paper). The JC-WT classification is a subsequent adaptation of the original Lamb Weather Types (LWT,

Lamb, 1972) approach to obtain an automated, objective method that is also applicable to other locations different from the British Isles. First of all, we load the required datasets, the NCEP-NCAR reanalysis1 (NCEP, Kalnay *et al.* 1996)), using to this aim the NOAA OpenDAP Service, and the package `loadR` for remote access. The NOAA OpenDAP Service is remotely accessible, without any login required. The `latLim` vector is used in the following to consider the Southern Hemisphere (SH) domain from 80° S to the equator as bounding box for data load. We consider only the year 2000 and the limited latitudes as an illustrative example.

```
latLim = c(-80, -0)
```

2.1 Loading reanalysis data

```
var <- "slp"
dataset <- "https://psl.noaa.gov/thredds/dodsC/Datasets/ncep.reanalysis/surface/slp.2
ncep <- loadGridData(dataset = dataset,
                      var = var,
                      latLim = latLim)
```

The function `clusterGrid` of package `transformerR` is the workhorse for the application of [clustering methods](#) to climate datasets. Here, we indicate the JC-WTs based on the Lamb Weather Typing through the argument `type = "lamb"`. The location of the WTs calculations is indicated with `center.point`. Additionally, other option is flag `typeU`, that is used to either include the U type in the calculations or not. We calculate JC-WTs for two locations: $p1 = [-70, -55]$ and $p2 = [-135, -15]$.

```

p1 <- c(-70, -55)
p2 <- c(-135, -15)

wts.ncep.p1 <- clusterGrid(ncep, type = "lamb", center.point = p1, typeU = TRUE)
wts.ncep.p2 <- clusterGrid(ncep, type = "lamb", center.point = p2, typeU = TRUE)

## Figure 1: SpatialPlot of annual climatologies from the 27 WTs for NCEP at p1:
wts.names <- c("A", "ANE", "AE", "ASE", "AS", "ASW", "AW", "ANW", "AN",
               "NE", "E", "SE", "S", "SW", "W", "NW", "N",
               "C", "CNE", "CE", "CSE", "CS", "CSW", "CW", "CNW", "CN", "U")

#grid of points from the 16-points "cross" used for the "p1" locations:
centerlon.p1 = -70
centerlat.p1 = -55
lon.array.p1 <- rep(centerlon.p1, times = 16) + c(-5, 5, -15, -5, 5, 15, -15, -5, 5,
                                                  -15, -5, 5, 15, -5, 5)
lat.array.p1 <- rep(centerlat.p1, times = 16) + c(10, 10, 5, 5, 5, 5, 0, 0, 0, 0,
                                                  -5, -5, -5, -5, -10, -10)
coords.p1 <- cbind(lon.array.p1, lat.array.p1) %>% sp::SpatialPoints()
l.points.p1 <- list("sp.points", coords.p1, col = 1)

#center 1 JC-WT frequencies and WTs climatologies
freqJCWT.p1 <- getWT(wts.ncep.p1) %>% table() %>% prop.table()
#To set freqJCWTs[i] = 0 when JCWTs 'i' does not occur:
freqJCWT.p1 <- freqJCWT.p1[match(1:27, names(freqJCWT.p1))]
freqJCWT.p1[which(is.na(freqJCWT.p1))] <- 0

```

```

names(freqJCWT.p1) <- wt.names
freqJCWT.p1 <- round(freqJCWT.p1 * 100, 2)
names.attr <- paste0(wt.names, ": ", freqJCWT.p1[wt.names], "%")
JCWTs.list <- lapply(1:27, function(x) {
  suppressMessages(climatology(subsetGrid(wts.ncep.p1, cluster = x)))
})
JCWTs.mg <- makeMultiGrid(JCWTs.list, skip.temporal.check = TRUE)

```

#center 2 JC-WT frequencies

```

freqJCWT.p2 <- getWT(wts.ncep.p2) %>% table() %>% prop.table()
#To set freqJCWTs[i] = 0 when JCWTs 'i' does not occur:
freqJCWT.p2 <- freqJCWT.p2[match(1:27, names(freqJCWT.p2))]
freqJCWT.p2[which(is.na(freqJCWT.p2))] <- 0
names(freqJCWT.p2) <- wt.names
freqJCWT.p2 <- round(freqJCWT.p2 * 100, 2)

```

```
dev.new()
```

```
breaks <- seq(99300, 103300, 200)
```

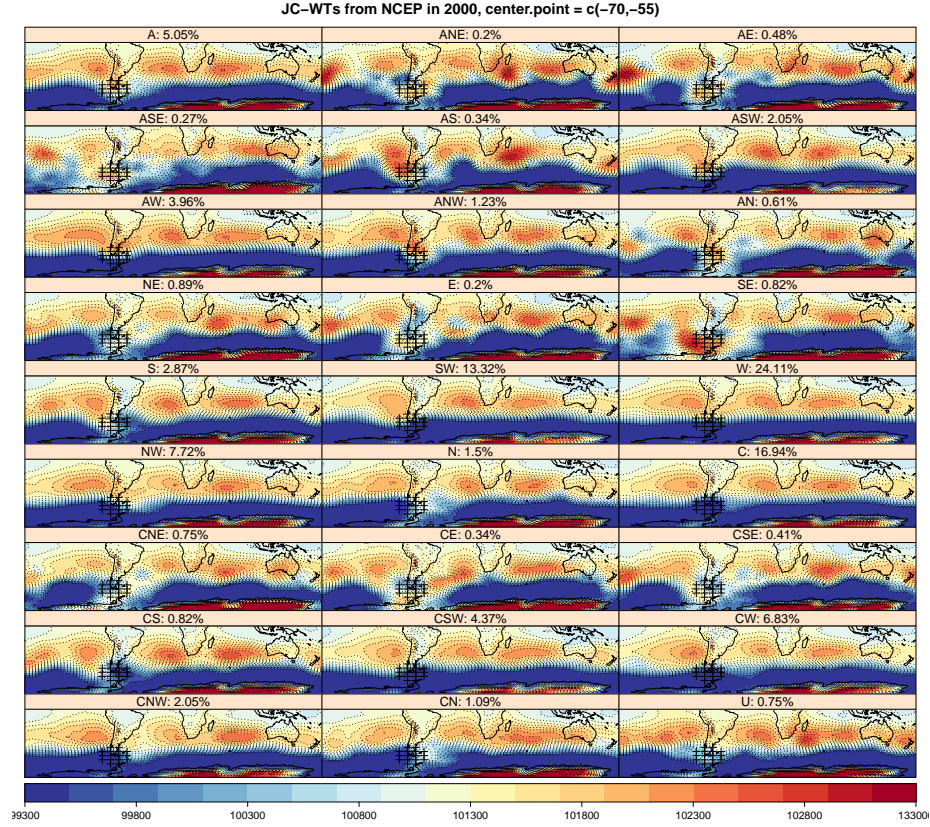
```
colorkey.labels <- c(99300, 99800, 100300, 100800, 101300, 101800, 102300, 102800, 103300)
```

The function `spatialPlot` from package `visualizeR` (Frías *et al.* 2018) is a wrapper for the `spplot` method in package `sp`, thus accepting the many possible arguments of the lattice framework for fine tuning of the plot. Here, we illustrate the annual climatologies from the 27 JC-WTs for NCEP at p1, in a similar way as the Fig. 1 from Fernandez-Granja *et*

al. (2021) in order to show the good adaptation of JC-WTs equations in Southern Hemisphere:

```
visualizeR::spatialPlot(JCWTs.mg,
  sp.layout = list(l.points.p1),
  backdrop.theme = "coastline",
  rev.colors = TRUE,
  main = "JC-WTs from NCEP in 2000, center.point = c(-70,-55)",
  useRaster = TRUE,
  set.min = min(breaks),
  set.max = max(breaks),
  at = breaks,
  colorkey = list(space = 'bottom',
    labels = list(at = seq(99300, 103300, 500),
      labels = colorkey.labels)),
  layout = c(3,9),
  as.table = TRUE,
  names.attr = names.attr,
  contour = TRUE,
  lty = 3)

\begin{figure}
```



{

}

\caption{Composite maps of the 27 JC-WTs derived from 6-hourly SLP (Pa) from the NCEP reanalysis for the year 2000. Sub-panels are labelled with their LWT abbreviation (frequency in % in parenthesis). Colorbar is centered on average sea-level atmospheric pressure (reds are highs and blues are lows). Lamb's cross coordinates are also indicated

over the Cape Horn domain, choosen as “p1” in this working example}

`\end{figure}`

Next, the WT frequencies as captured by the NCEP reanalysis at “p1” and “p2” are also displayed as a barplot. With this frequencies profiles, we will be able to analyze two factors: number of WTs observed and the Unclassified (U) type relative frequency. This factors are taken into account in order to assess the suitability of JC-WTs method at any location, as done in Fig. 3 from this paper. In this example, the assessment is done only for 2 single locations, instead of for the whole world (see Fig. 3 from the paper).

```
wts.ncep.freqs <- matrix(c(freqJCWT.p1, freqJCWT.p2),
                          ncol = length(wt.names),
                          byrow = TRUE,
                          dimnames = list(c("p1", "p2"), wt.names))

layout(matrix(c(rep(1, 6), 2), ncol = 1))
par(mai = rep(0.6, 4))
bar.colors <- c("blue", "red")
bp <- barplot(wts.ncep.freqs,
              beside = TRUE, col = bar.colors, border = NA, las = 1,
              ylab = "freq. [%]", ylim = c(0, 100),
              main = "NCEP-NCAR Reanalysis (NCEP) JC-WT frequencies at 2 locations")
at=seq(3.5, (3.5+27*1.5*2), 1.5*2)
abline(v = at, col = "gray33", lty = 3)
par(mai = c(0, 0, 0, 0))
```

```

plot.new()

legend(legend = c("p1: [70°W, 55°S]", "p2: [135°W, 12.5°S]") ,
      fill = c("blue", "red"),
      "center", horiz = TRUE, border = "transparent", bty = "n")

```

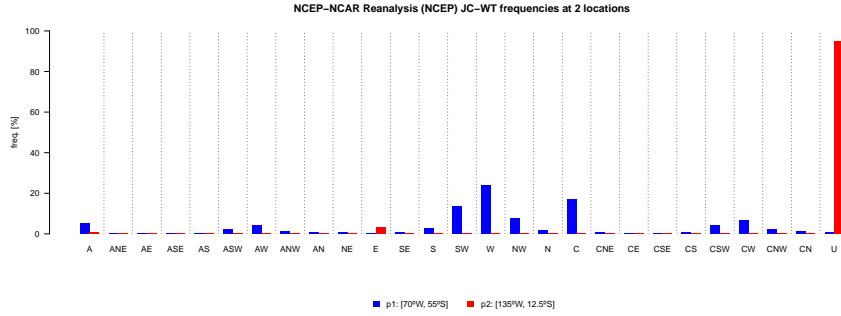


Figure 1: Comparison of JC-WTs relative frequencies profile for "p1" (blue) and "p2" (red) obtained from the NCEP reanalysis.

From the figure of above, we can observed that for “p1”, all JC-WTs are observed, and the U type relative frequency is low (0.75%). For “p2”, only 8 JC-WTs are observed, and the U type relative frequency is large (close to 95%). Thus, JC-WT classification is suitable for “p1” but not for “p2”.

3 Assessment of JC-WTs frequencies profiles for the different IPCC regions of the globe.

In this second example, the JC-WT frequencies profiles from all the grid-boxes contained in a single IPCC region for NCEP will be calculated, and the spatial variability of the WTs frequencies will be assessed. This is similarly done for Fig. 6 of the paper. For this

purpose, the JC-WT classification has to be performed for the entire region(s) of interest. Thus, our JC-WT dataset products can be used (<https://doi.org/10.5281/zenodo.5761257>). Reanalyses such as NCEP, JRA-55, ERA-Interim, ERA5 and ERA-20Care available.

3.1 Loading JC-WTs reanalysis products from Zenodo:

```
destfile <- tempfile()
download.file("https://zenodo.org/record/5761258/files/NCEP.nc?download=1", destfile

#di <- dataInventory(destfile)

JC.ncep <- loadGridData(dataset = destfile,
                        var = "JCWT",
                        years = 2000)
```

3.2 Post-processing with geoprocessor:

In order to spatially subset the grid to the desire IPCC region, we use the followings functions from package `geoprocessor` (<https://github.com/SantanderMetGroup/geoprocessor>), part of the `climate4R` framework (<https://github.com/SantanderMetGroup/climate4R>). Further details of `geoprocessor` features can be found in the WIKI section of the package (<https://github.com/SantanderMetGroup/geoprocessor/wiki>).

```
#Downloading the regions shapefiles:
```

```
tmpdir <- tempdir()
```

```
destfile <- tempfile()
```

```
download.file(url = "http://meteo.unican.es/work/fdezja/lamb_regions_IPCC/IPCC-WGI-re  
destfile = destfile)
```

```
unzip(destfile, exdir = tmpdir)
```

```
#Import vector layer:
```

```
refregions <- readOGR(dsn = paste(tmpdir, "IPCC-WGI-reference-regions-v4-ET_shapefile"
```

```
## OGR data source with driver: ESRI Shapefile
```

```
## Source: "/tmp/RtmpM8SyJU/IPCC-WGI-reference-regions-v4-ET_shapefile", layer: "IPCC"
```

```
## with 41 features
```

```
## It has 4 fields
```

```
refregions.data <- as(refregions, "SpatialPolygons")
```

```
JC.ncep <- projectGrid(JC.ncep, original.CRS = proj4string(refregions))
```

```
## Warning in projectGrid(JC.ncep, original.CRS = proj4string(refregions)):
```

```
## CAUTION! Grid with previously defined projection: LatLonProjection
```

The MED region is chosen for this example, corresponding to region #

12. The rest of the IPCC regions features can be found in the Fig.1 of

M. Iturbide *et al.* (2020, <https://doi.org/10.5194/essd-12-2959-2020>).

```
region.ID <- 12
```

```
JC.ncep.MED <- overGrid(JC.ncep, refregions.data[names(refregions.data)[[region.ID]]]
```

3.3 Frequency profiles from the grid-boxes of the IPCC region:

```
dim.grid <- dim(JC.ncep.MED$Data)
wts.table <- array(dim = c(27, dim.grid[2], dim.grid[3]))
for (i in 1:(dim.grid[2])) {
  for (j in 1:(dim.grid[3])) {
    wts.table.aux <- JC.ncep.MED$Data[ ,i,j] %>% table %>% prop.table
    wts.table.aux <- wts.table.aux[match(1:27, names(wts.table.aux))]
    wts.table.aux[which(is.na(wts.table.aux))] <- 0
    wts.table[ ,i,j] <- wts.table.aux
  }
}

wts.table[which(wts.table == 0)] <- NA
clim <- climatology(JC.ncep.MED)
JC.ncep.MED.list <- lapply(1:27, function(x) {
  wts.table[x, , ]
  clim$Data <- array(wts.table[x, , ], dim = c(1, dim(wts.table[x, , ])))
  attr(clim$Data, "dimensions") <- c("time", "lat", "lon")
  return(clim)
})

visualizeR::violinPlot(JC.ncep.MED.list,
  violin = FALSE, color.fun = list(FUN = median, na.rm = TRUE),
  color.theme = "YlOrRd",
  h.lines = seq(0, 0.4, 0.05),
```

```

bwplot.custom = list(main = "Dispersion JC-WTs relative frequen

ylim = c(0, 0.4),

do.out = FALSE,

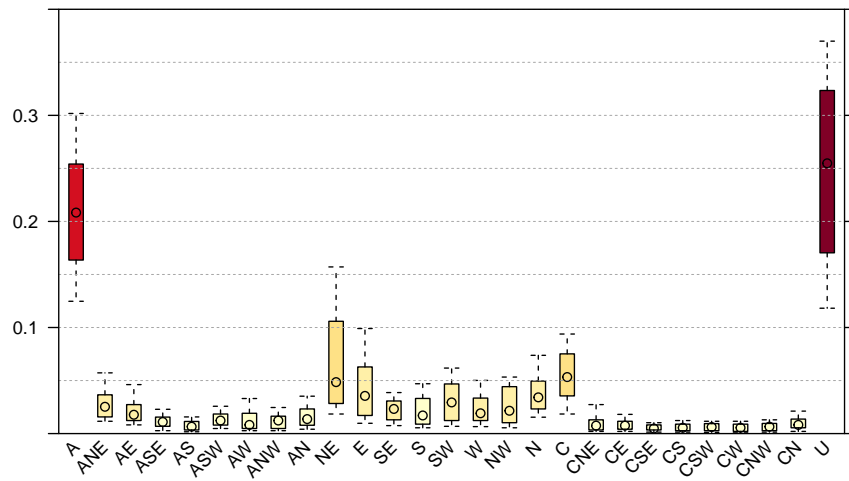
do.conf = FALSE,

scales=list(x=list(rot=45,

labels = wt.names))))

```

Dispersion JC–WTs relative frequencies in MED* region (NCEP, year = 2000)



We can see in the Figure from above the spatial variability of the WTs from MED region for NCEP in year 2000. These results are also shown in the Fig. 6 from the paper. The remaining result from this paper (not illustrated in this notebook) can be obtained with the help of the

notebook from Fernandez-Granja *et al.* (2021,

https://github.com/SantanderMetGroup/notebooks/blob/devel/2020_Lamb_ClimDyn.pdf). The basics for the calculations needed to obtain

the TPMS from Figs. 3 and 4 of this paper are illustrated there.

4 References

- Fernandez-Granja, J. A., Casanueva, A., Bedia, J., and Fernández, J.: Improved atmospheric circulation over Europe by the new generation of CMIP6 earth system models, *Climate Dynamics*, 56, <https://doi.org/10.1007/s00382-021-05652-9>, 2021.
- Frías, M.D., Iturbide, M., Manzanas, R., Bedia, J., Fernández, J., Herrera, S., Cofiño, A.S., Gutiérrez, J.M., 2018. An R package to visualize and communicate uncertainty in seasonal climate prediction. *Environmental Modelling & Software* 99, 101–110. <https://doi.org/10.1016/j.envsoft.2017.09.008>
- Iturbide, M., Bedia, J., Herrera, S., Baño-Medina, J., Fernández, J., Frías, M.D., Manzanas, R., San-Martín, D., Gimadevilla, E., Cofiño, A.S., Gutiérrez, J.M., 2019. The R-based climate4R open framework for reproducible climate data access and post-processing. *Environmental Modelling & Software* 111, 42–54. <https://doi.org/10.1016/j.envsoft.2018.09.009>
- Iturbide, M., Gutiérrez, J., Alves, L., Bedia, J., Cerezo-Mota, R., Gimadevilla, Ezequiel, Cofiño, Antonio, Di Luca, Alejandro, Faria, Sergio, Gorodetskaya, Irina, Hauser, Mathias, Herrera García, Sixto, Hennessy, Kevin, Hewitt, Helene, Jones, R., Krakovska, Svitlana, Manzanas, Rodrigo, Martínez-Castro, Daniel, Narisma, Gemma, Vera, Carolina. (2020). An update of IPCC climate reference re-

gions for subcontinental analysis of climate model data: definition and aggregated datasets. *Earth System Science Data*. 12. 2959-2970. <https://doi.org/10.5194/essd-12-2959-2020>.

- Jenkinson, A. and Collison, F.: An initial climatology of gales over the North sea. Synoptic Climatology Branch Memorandum, Meteorological Office, 62, 1977.
- Lamb, H.: British Isles Weather types and a register of daily sequence of circulation patterns 1861-1971., Meteorological Office, Geophysical Memoir., 116, 1-85, 1972.
- Milton Bache, S. and Wickham, H., 2014. magrittr: A Forward-Pipe Operator for R. R package version 1.5. <https://CRAN.R-project.org/package=magrittr>
- Pebesma, E.J., R.S. Bivand, 2005. Classes and methods for spatial data in R. R News 5 (2), <https://cran.r-project.org/doc/Rnews/>.
- Roger S. Bivand, Edzer Pebesma, Virgilio Gomez-Rubio, 2013. Applied spatial data analysis with R, Second edition. Springer, NY. <https://asdar-book.org/>
- Sarkar, Deepayan, 2008. Lattice: Multivariate Data Visualization with R. Springer, New York. ISBN 978-0-387-75968-5
- Wickham, H., Hester, J. and Chang, W., 2020. devtools: Tools to Make Developing R Packages Easier. R package version 2.3.0. <https://CRAN.R-project.org/package=devtools>

5 Session info

```
sessionInfo() %>% print()

## R version 3.6.0 (2019-04-26)

## Platform: x86_64-pc-linux-gnu (64-bit)

## Running under: Ubuntu 18.04.2 LTS

##

## Matrix products: default

## BLAS:   /usr/lib/x86_64-linux-gnu/blas/libblas.so.3.7.1

## LAPACK: /usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.7.1

##

## locale:
##  [1] LC_CTYPE=es_ES.UTF-8          LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8          LC_COLLATE=es_ES.UTF-8
##  [5] LC_MONETARY=en_US.UTF-8      LC_MESSAGES=es_ES.UTF-8
##  [7] LC_PAPER=en_US.UTF-8         LC_NAME=en_US.UTF-8
##  [9] LC_ADDRESS=en_US.UTF-8       LC_TELEPHONE=en_US.UTF-8
## [11] LC_MEASUREMENT=en_US.UTF-8   LC_IDENTIFICATION=en_US.UTF-8
##

## attached base packages:
## [1] stats      graphics  grDevices  utils      datasets  methods   base

##

## other attached packages:
## [1] magrittr_1.5      geoprocessor_0.2.0 rgdal_1.5-28      sp_1.3-1
## [5] visualizeR_1.6.1  transformerR_2.1.3 loader_1.4.15      loader.java_1.1.1
```

```
## [9] rJava_0.9-11
##
## loaded via a namespace (and not attached):
## [1] Rcpp_1.0.2          lattice_0.20-38      zoo_1.8-6
## [4] digest_0.6.20       foreach_1.4.4        SpecsVerification_0.5-2
## [7] CircStats_0.2-6     evaluate_0.14        spam_2.2-2
## [10] rlang_0.4.0         data.table_1.12.2    raster_3.0-2
## [13] R.oo_1.24.0         R.utils_2.10.1       Matrix_1.2-17
## [16] rmarkdown_2.1       stringr_1.4.0        RcppEigen_0.3.3.5.0
## [19] RCurl_1.95-4.12     munsell_0.5.0        proxy_0.4-23
## [22] compiler_3.6.0      easyVerification_0.4.4 xfun_0.12
## [25] htmltools_0.4.0     tcltk_3.6.0          dtw_1.20-1
## [28] codetools_0.2-16    mapplots_1.5.1       MASS_7.3-51.4
## [31] bitops_1.0-6        R.methodsS3_1.8.1    grid_3.6.0
## [34] scales_1.0.0        stringi_1.4.3        sm_2.2-5.6
## [37] pbapply_1.4-1       kohonen_3.0.8        latticeExtra_0.6-28
## [40] akima_0.6-2         padr_0.5.0           vioplot_0.3.6
## [43] boot_1.3-22         RColorBrewer_1.1-2   iterators_1.0.10
## [46] tools_3.6.0         gdalUtils_2.0.3.2    maps_3.3.0
## [49] fields_9.8-6        abind_1.4-5          parallel_3.6.0
## [52] yaml_2.2.0          colorspace_1.4-1     verification_1.42
## [55] dotCall64_1.0-0     knitr_1.28
```