UNIWERSYTET PRZYRODNICZY WE WROCŁAWIU

Pakiet R *climate* jako narzędzie do automatyzacji analiz klimatycznych

mgr. inż Arkadiusz Głogowski

22.06.22r. Wrocław Seminarium środowiskowe IMGW-PIB i UPWR

sustainability

MDPI

bczernecki Bartosz Czernecki

Nowosad Jakub Nowosad

aglogowski Arkadiusz Głogowski

kadyb Krzysztof Dyba

Technical Note

Climate: An R Package to Access Free In-Situ Meteorological and Hydrological Datasets For Environmental Assessment

Bartosz Czernecki [1], Arkadiusz Głogowski [2,*] and Jakub Nowosad [3]

https://doi.org/10.3390/su12010394

Article Metrics

Citations

Crossref	Web of Science	Scopus
21	17	18

Stan na dzień 20.06.2022

climate

build passing | R-CMD-check passing

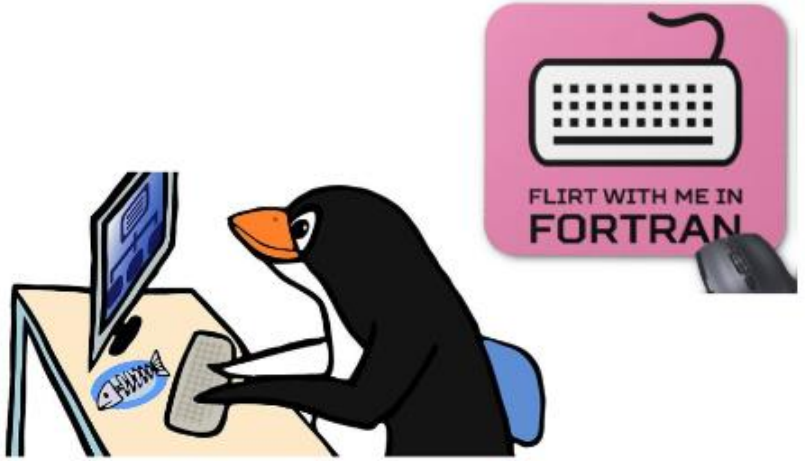
CRAN 1.0.4 | downloads 994/month | downloads 32K

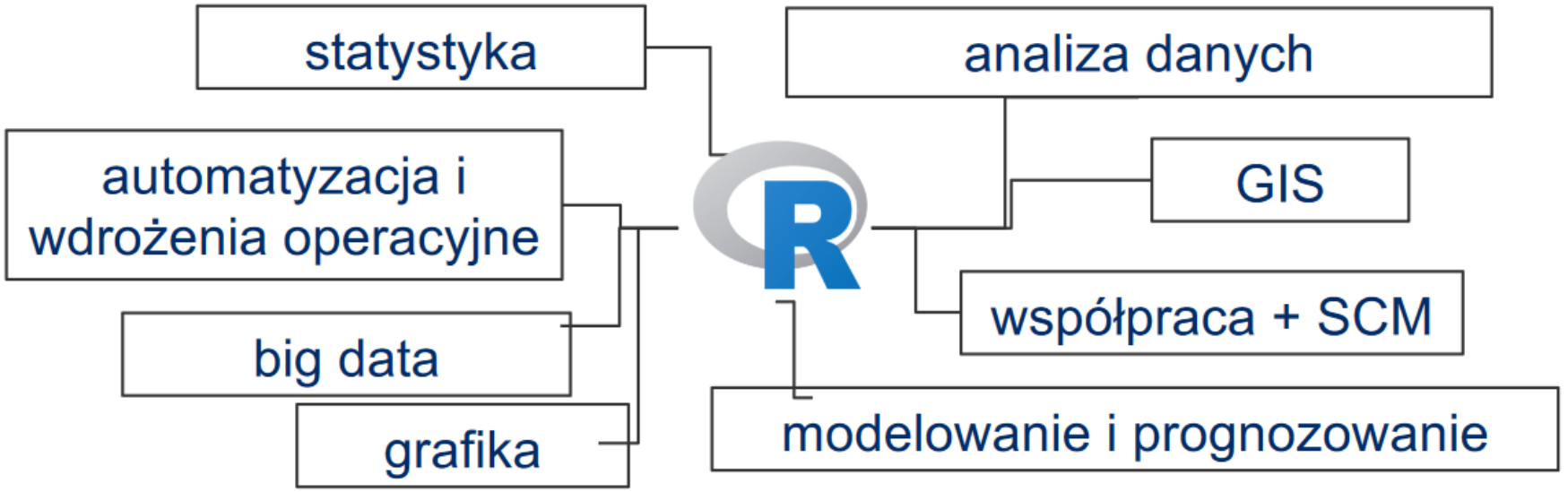
The goal of the **climate** R package is to automatize downloading of meteorological and hydrological data from publicly available repositories:

- OGIMET (ogimet.com)
- University of Wyoming - atmospheric vertical profiling data (http://weather.uwyo.edu/upperair/)
- National Oceanic & Atmospheric Administration - Earth System Research Laboratories - Global Monitoring Laboratory (NOAA)
- Polish Institute of Meterology and Water Management - National Research Institute (IMGW-PIB)
- National Oceanic & Atmospheric Administration - National Climatic Data Center - Integrated Surface Hourly (ISH) (NOAA)

Czym jest język R

- Odkrywanie nowej wiedzy wymaga pracy na danych
- Ważnym czynnikiem przy używaniu danych jest ich dostępność i przystępność w korzystaniu
- Różne problemy = wiele programów **!= automatyzacja**





Po co nam pakiet R climate ?

Cel:

Automatyzacja wyszukiwania i pobierania zbiorów danych meteorologicznych i hydrologicznych z ogólnodostępnych repozytoriów danych

Główne funkcjonalności pakietu climate

Meteorological data

- **meteo_ogimet()** - Downloading hourly and daily meteorological data from the SYNOP stations available in the ogimet.com collection. Any meteorological (aka SYNOP) station working under the World Meteorological Organizaton framework after year 2000 should be accessible.
- **meteo_imgw()** - Downloading hourly, daily, and monthly meteorological data from the SYNOP/CLIMATE/PRECIP stations available in the danepubliczne.imgw.pl collection. It is a wrapper for `meteo_monthly()`, `meteo_daily()`, and `meteo_hourly()` from the **imgw** package.
- **meteo_noaa_hourly()** - Downloading hourly NOAA Integrated Surface Hourly (ISH) meteorological data - Some stations have > 100 years long history of observations
- **sounding_wyoming()** - Downloading measurements of the vertical profile of atmosphere (aka rawinsonde data)
- **meteo_noaa_co2()** - Downloading monthly CO2 measurements from Mauna Loa Observatory

Hydrological data

- **hydro_imgw()** - Downloading hourly, daily, and monthly hydrological data from the SYNOP / CLIMATE / PRECIP stations available in the danepubliczne.imgw.pl collection. It is a wrapper for `hydro_annual()`, `hydro_monthly()`, and `hydro_daily()` from the **imgw** package.

Pomocnicze funkcjonalności pakietu climate

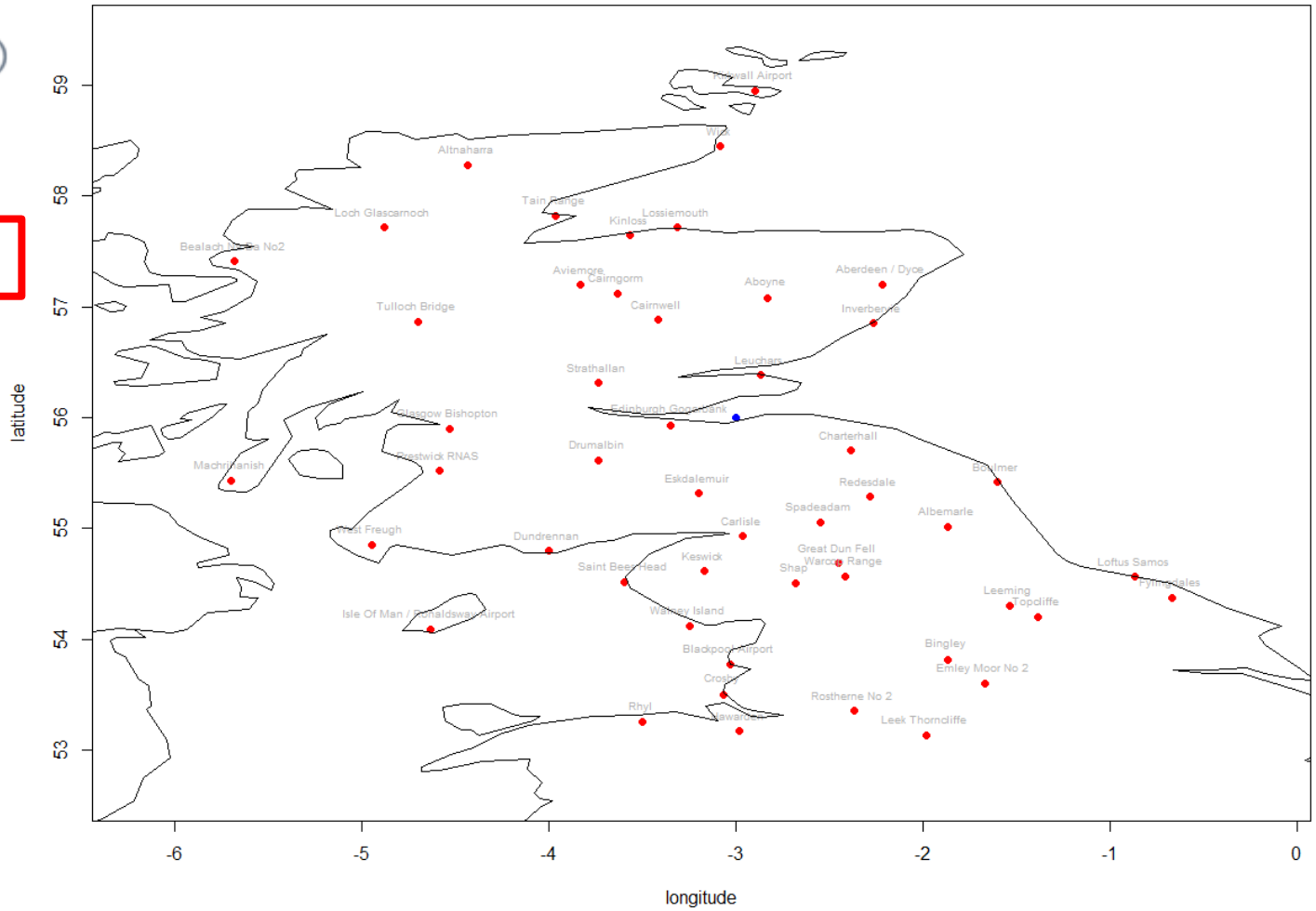
Auxiliary functions and datasets

- **stations_ogimet()** - Downloading information about all stations available in the selected country in the Ogimet repository
- **nearest_stations_ogimet()** - Downloading information about nearest stations to the selected point using Ogimet repository
- **nearest_stations_noaa()** - Downloading information about nearest stations to the selected point available for the selected country in the NOAA ISH meteorological repository
- **nearest_stations_imgw()** - List of nearby meteorological or hydrological IMGW-PIB stations in Poland
- **imgw_meteo_stations** - Built-in metadata from the IMGW-PIB repository for meteorological stations, their geographical coordinates, and ID numbers
- **imgw_hydro_stations** - Built-in metadata from the IMGW-PIB repository for hydrological stations, their geographical coordinates, and ID numbers
- **imgw_meteo_abbrev** - Dictionary explaining variables available for meteorological stations (from the IMGW-PIB repository)
- **imgw_hydro_abbrev** - Dictionary explaining variables available for hydrological stations (from the IMGW-PIB repository)

Nie wiemy nic o obszarze??

```
ns=nearest_stations_ogimet(country = "United+Kingdom",
                           add_map = T,
                           no_of_stations = 50,
                           point = c(-3,56),
                           date = Sys.Date())
```

	wmo_id	station_names	lon	lat	alt	distance [km]
33	03166	Edinburgh Gogarbank	-3.350007	55.93335	57	39.97523
34	03171	Leuchars	-2.866684	56.38334	12	45.53650
31	03158	Charterhall	-2.383341	55.70001	112	76.93943
32	03162	Eskdalemuir	-3.200004	55.31667	242	79.88351
28	03144	Strathallan	-3.733348	56.31667	35	89.62268
30	03155	Drumalbin	-3.733348	55.61668	245	92.84126
17	03072	Cairnwell	-3.416675	56.88335	933	109.58158
44	03230	Redesdale	-2.283339	55.28334	211	113.71264
40	03224	Spadeadam	-2.550011	55.05000	285	117.93937
39	03220	Carlisle	-2.966686	54.93335	28	119.73271
19	03080	Aboyne	-2.833350	57.08333	140	122.97630



Przykładowe dane 1

```
edynburg=meteo_ogimet(interval = "daily",
                      date = c("2021-12-01","2022-05-30"),
                      station = "03166",
                      coords = T)
```

interval:

- daily
- hourly

	station_ID	Lon	Lat	Date	TemperatureCAvg	TemperatureCMax	TemperatureCMin	TdAvgC	HrAvg	WindkmhDir	WindkmhInt	WindkmhGust	PresslevHp	Precmm
3	3166	-3.35	55.93333	2022-05-30	10.2	14.9	4.3	5.6	74.7	WNW	7.9	29.7	1017.4	1.4
4	3166	-3.35	55.93333	2022-05-29	11.2	16.7	4.3	4.7	65.8	NNE	9.6	25.9	1025.8	0.0
5	3166	-3.35	55.93333	2022-05-28	11.1	15.5	6.5	5.8	70.7	W	21.5	63.0	1023.3	0.0
6	3166	-3.35	55.93333	2022-05-27	10.7	15.3	7.4	5.9	73.0	WSW	29.3	64.9	1013.9	1.0
7	3166	-3.35	55.93333	2022-05-26	11.3	15.3	7.9	7.2	76.6	WSW	23.6	66.7	1005.9	1.4
8	3166	-3.35	55.93333	2022-05-25	11.9	16.5	8.2	6.9	71.5	WSW	16.9	40.8	1001.9	3.4

```
#można zapisać do arkusza kalkulacyjnego:
writexl::write_xlsx(edynburg, "excel.xlsx")
```

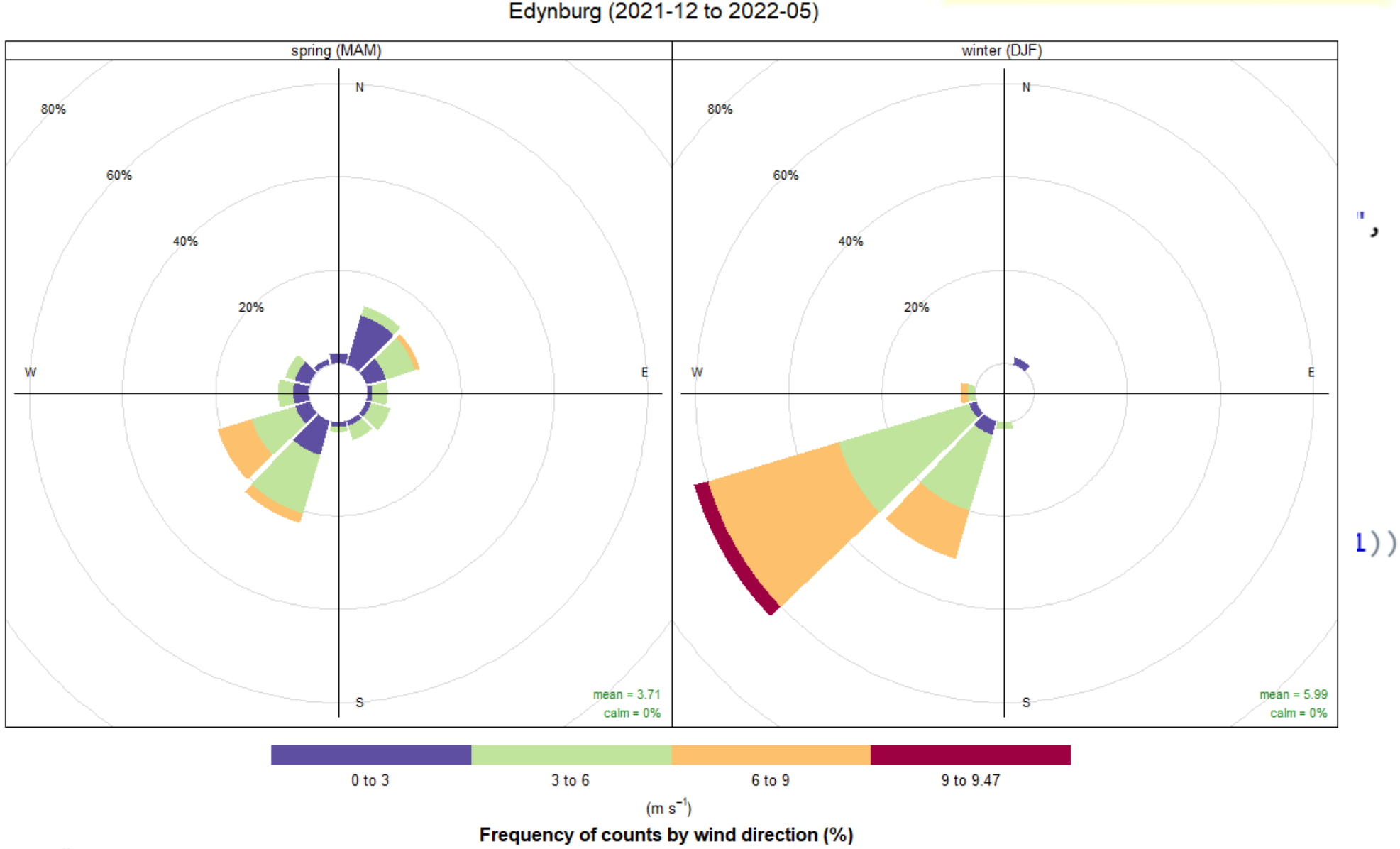
A1 | station_ID

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	station_ID	Lon	Lat	Date	perature	peratureC	perature	TdAvgC	HrAvg	VindkmhD	VindkmhIn	indkmhGu	PresslevHp	Precmm	TotClOct	lowClOct	SunD1h	VisKm	PreselevHp	nowDepcm	
2	3166	-3.35	55.93333	2022-05-30	10.2	14.9	4.3	5.6	74.7	WNW	7.9	29.7	1017.4	1.4	8	8	0.8	32			
3	3166	-3.35	55.93333	2022-05-29	11.2	16.7	4.3	4.7	65.8	NNE	9.6	25.9	1025.8	0	6.9	7.2	6.0	39.4			
4	3166	-3.35	55.93333	2022-05-28	11.1	15.5	6.5	5.8	70.7	W	21.5	63	1023.3	0	5.5	5.7	6.4	35.1			
5	3166	-3.35	55.93333	2022-05-27	10.7	15.3	7.4	5.9	73	WSW	29.3	64.9	1013.9	1	5.2	5.9	6.2	32			
6	3166	-3.35	55.93333	2022-05-26	11.3	15.3	7.9	7.2	76.6	WSW	23.6	66.7	1005.9	1.4	6.3	6.6	5.3	42.3			
7	3166	-3.35	55.93333	2022-05-25	11.9	16.5	8.2	6.9	71.5	WSW	16.9	40.8	1001.9	3.4	7.2	7.2	6.2	52.4			

Obróbka danych w R – przykład 1 róża wiatru

```
librar
# conv
wdir <


# chan
edynbu
edynbu
edynbu
edynbu
windRo
```

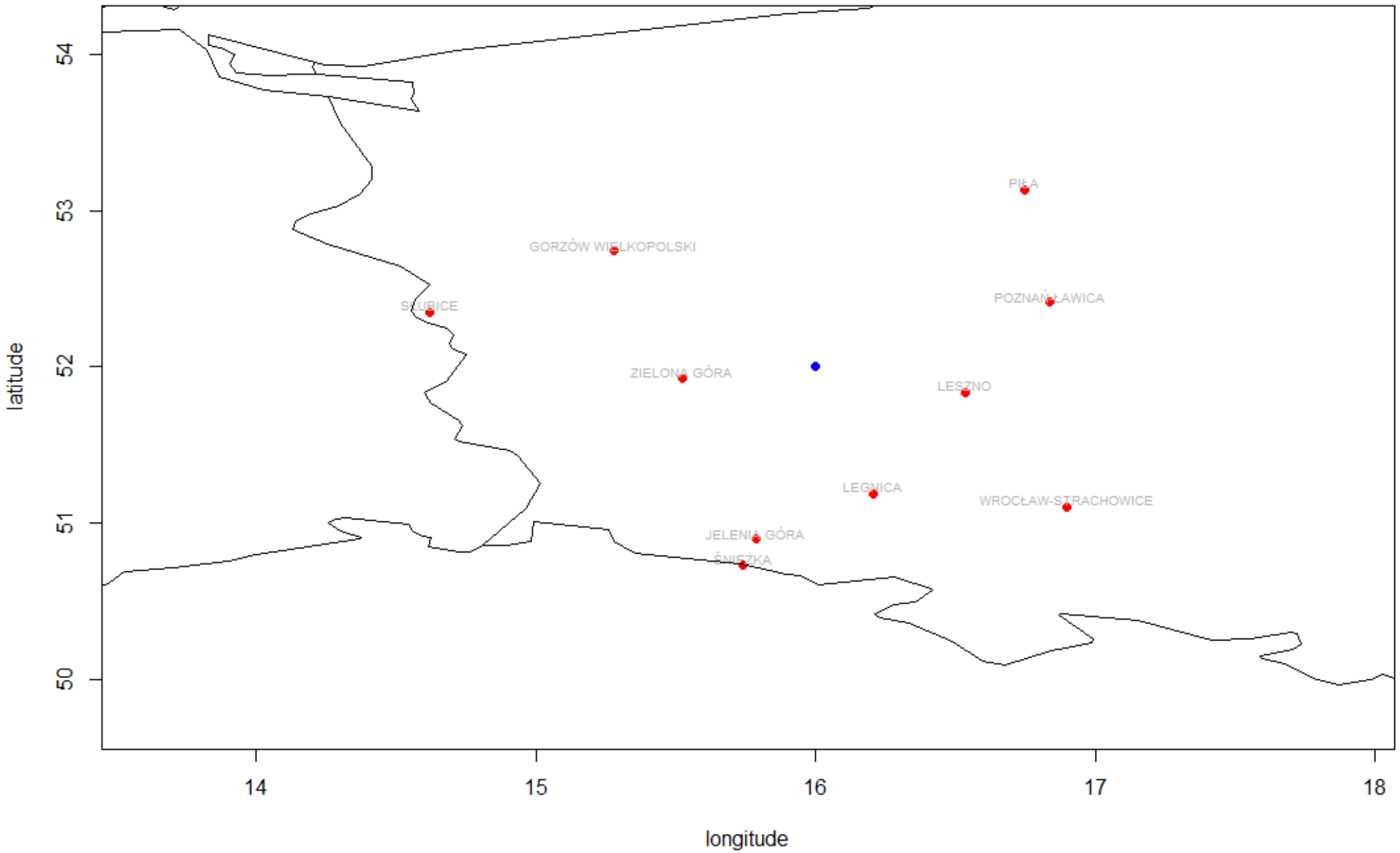
```
",

1))
```



Przykładc

```
ns2=nearest
```

	id
252	351150400
271	351160418
401	352160330
257	351160415
388	352150300
383	352140310
277	351160424
89	350150500
503	353160230
103	350150510



Nearest_station_imgw

```
ns2=nearest_stations_imgw(type = "meteo",
                          rank = "synop",
                          year = 2021,
                          add_map = T,
                          point = c(16,52),
                          no_of_stations =10 )
```

type:

1. **Meteo**
2. **Hydro**

Name	Last modified	Size	Description
Parent Directory		-	
Biuletyn_PSHM/	2022-06-21 07:00	-	
Roczniki/	2020-06-30 23:13	-	
dane_aktynometryczne/	2018-02-23 10:27	-	
dane_hydrologiczne/	2022-04-11 15:16	-	
dane_meteorologiczne/	2017-11-27 13:25	-	
test/	2020-05-03 22:34	-	

rank:

1. **synop**
2. **climate**
3. **precip**

Name	Last modified	Size	Description
Parent Directory		-	
klimat/	2022-03-11 07:24	-	
opad/	2022-03-11 07:25	-	
synop/	2022-02-04 08:18	-	

Pobieranie danych meteo_imgw()

```
legnica=meteo_imgw(interval = "monthly",
                   rank = "synop",
                   year = 2020,
                   station = "LEGNICA",
                   coords = T,
                   col_names = "short")
```

legnica | 12 obs. of 44 variables

interval:

1. monthly
2. daily
3. hourly

rank:

1. synop
2. climate
3. precip

col_names:

1. short
2. long
3. polish

rank	id	X	Y	station	yy	mm	tmax_abs	tmax_mean	tmin_abs	tmin_mean	t2m_mean_mon	t5cm_min	rr_monthly	rr_max_daily	first_day_max_rr
SYNOPTYCZNA	351160415	16.20766	51.19252	LEGNICA	2020	1	13.2	6.8	-7.7	-0.4	3.1	-9.1	7.4	4.0	8
SYNOPTYCZNA	351160415	16.20766	51.19252	LEGNICA	2020	2	15.4	9.3	-2.9	2.6	5.7	-7.8	54.8	12.1	23
SYNOPTYCZNA	351160415	16.20766	51.19252	LEGNICA	2020	3	18.2	10.8	-7.4	0.5	5.2	-9.2	21.1	5.5	3
SYNOPTYCZNA	351160415	16.20766	51.19252	LEGNICA	2020	4	23.5	17.3	-5.6	1.9	9.9	-7.9	5.9	3.2	13
SYNOPTYCZNA	351160415	16.20766	51.19252	LEGNICA	2020	5	22.8	17.8	1.6	6.3	12.1	-0.8	62.9	20.1	11
SYNOPTYCZNA	351160415	16.20766	51.19252	LEGNICA	2020	6	31.0	22.9	6.3	12.7	17.6	4.6	108.7	20.1	24
SYNOPTYCZNA	351160415	16.20766	51.19252	LEGNICA	2020	7	34.3	26.0	6.9	12.7	19.2	4.5	26.3	11.9	16
SYNOPTYCZNA	351160415	16.20766	51.19252	LEGNICA	2020	8	32.8	26.9	8.8	15.0	20.7	6.4	96.1	26.8	30
SYNOPTYCZNA	351160415	16.20766	51.19252	LEGNICA	2020	9	30.5	21.8	3.8	10.4	15.8	0.8	69.2	25.1	25
SYNOPTYCZNA	351160415	16.20766	51.19252	LEGNICA	2020	10	24.0	14.7	1.1	8.7	11.4	-1.2	100.7	43.3	13
SYNOPTYCZNA	351160415	16.20766	51.19252	LEGNICA	2020	11	18.3	9.7	-4.7	3.4	6.4	-7.8	12.9	4.9	3
SYNOPTYCZNA	351160415	16.20766	51.19252	LEGNICA	2020	12	12.7	6.2	-5.3	0.0	2.9	-7.6	10.2	4.0	22

Pobieranie danych hydro_imgw()

```
obra=hydro_imgw(interval = "monthly",
                year = 2020,
                value = "H",
                station = "OBRA",
                coords = T,
                col_names = "short")
```

value:

1. H
2. Q
3. T

interval:

1. daily
2. monthly
3. semiannual_and_annual

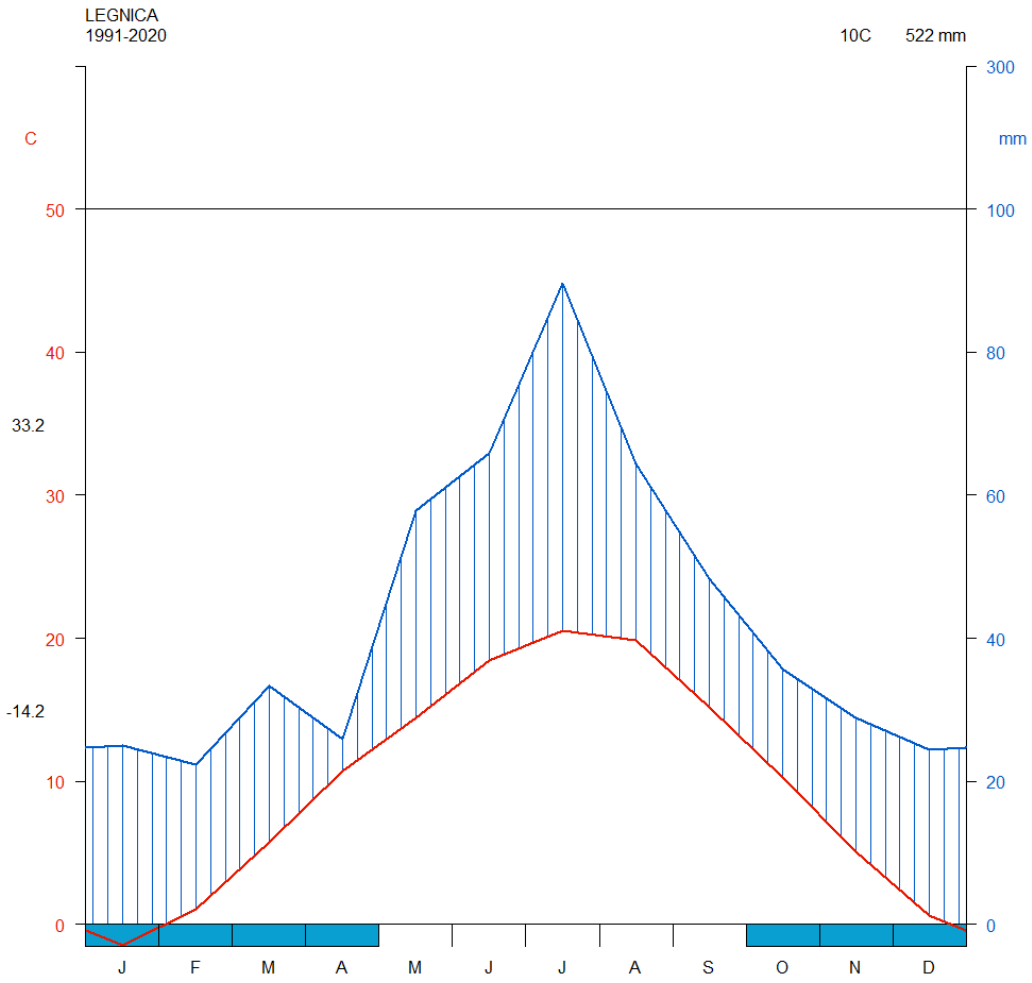
obra 36 obs. of 12 variables

	id	X	Y	station	riv_or_lake	hyy	idhyy	idex	H	Q	T	mm
16954	152160030	16.07617	52.07467	OBRA	Dojca (18782)	2020	1	1	259	0.43	NA	11
16955	152160030	16.07617	52.07467	OBRA	Dojca (18782)	2020	1	2	261	0.50	NA	11
16956	152160030	16.07617	52.07467	OBRA	Dojca (18782)	2020	1	3	267	0.62	NA	11
16957	152160030	16.07617	52.07467	OBRA	Dojca (18782)	2020	2	1	254	0.39	NA	12
16958	152160030	16.07617	52.07467	OBRA	Dojca (18782)	2020	2	2	258	0.49	NA	12
16959	152160030	16.07617	52.07467	OBRA	Dojca (18782)	2020	2	3	261	0.61	NA	12
16960	152160030	16.07617	52.07467	OBRA	Dojca (18782)	2020	3	1	246	0.18	NA	1
16961	152160030	16.07617	52.07467	OBRA	Dojca (18782)	2020	3	2	254	0.38	NA	1
16962	152160030	16.07617	52.07467	OBRA	Dojca (18782)	2020	3	3	260	0.58	NA	1
16963	152160030	16.07617	52.07467	OBRA	Dojca (18782)	2020	4	1	248	0.15	NA	2

Przykład 2 – dane IMGW

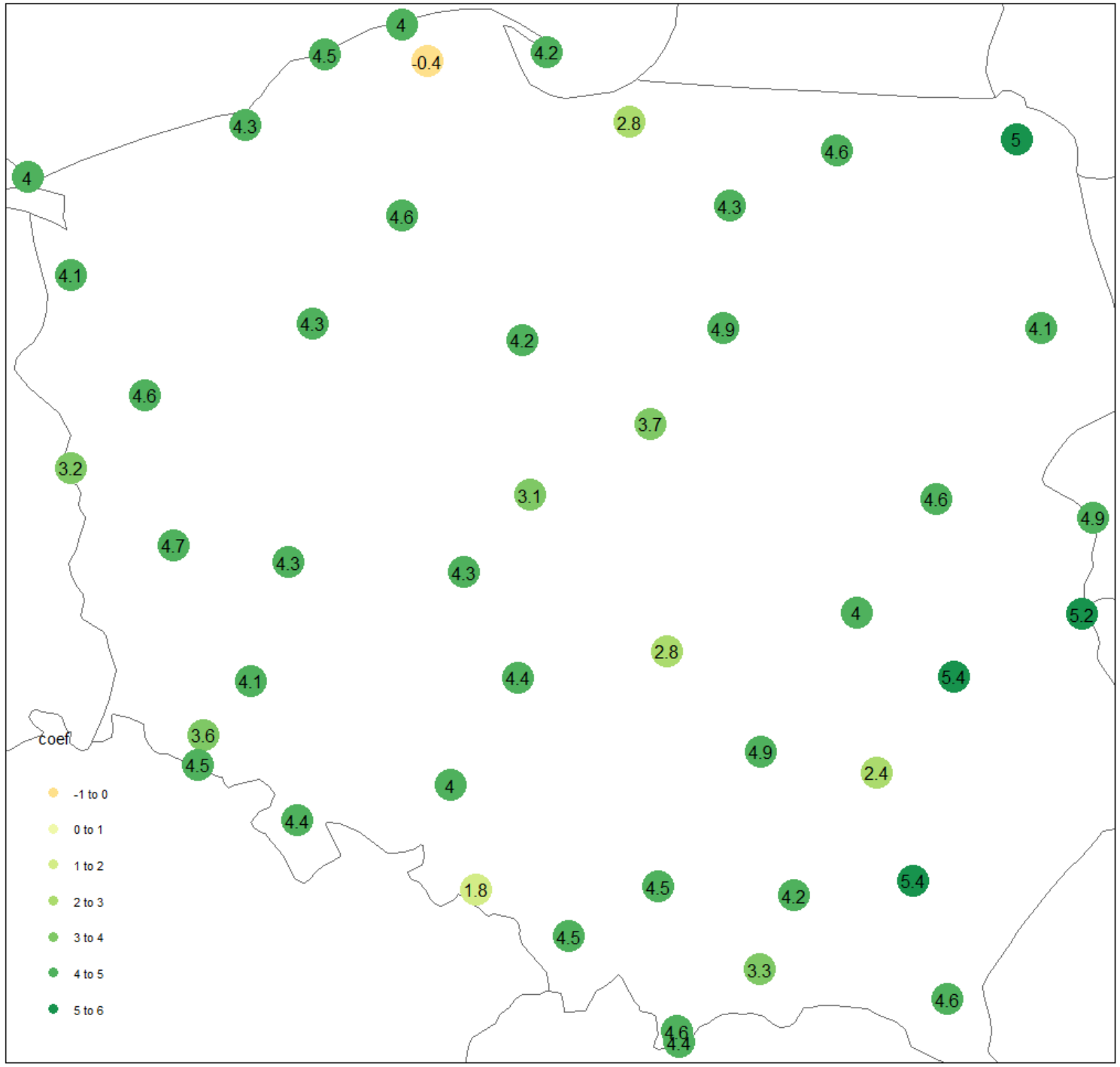
```
install.packages("climatol")
library(climatol)
legnica=meteo_imgw(interval = "monthly",
                   rank = "synop",
                   year = 1991:2020,
                   station = "LEGNICA",
                   coords = T,
                   col_names = "short")
df2 = select(legnica, station:t2m_mean_mon, rr_monthly)

monthly_summary = df2 %>%
  group_by(mm) %>%
  summarise(tmax = mean(tmax_abs, na.rm = TRUE),
            tmin = mean(tmin_abs, na.rm = TRUE),
            tavg = mean(t2m_mean_mon, na.rm = TRUE),
            prec = sum(rr_monthly) / n_distinct(yy))

monthly_summary = as.data.frame(t(monthly_summary[, c(5,2,3,4)]))
monthly_summary = round(monthly_summary, 1)
colnames(monthly_summary) = month.abb
print(monthly_summary)
# create plot with use of the "climatol" package:
climatol::diagwl(monthly_summary, mlab = "en",
                 est = "POZNAŃ", alt = NA,
                 per = "1991-2019", p3line = FALSE)
```

```
> print(monthly_summary)
       Jan   Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov   Dec
prec  25.0  22.3 33.3 25.9 57.8 65.9 89.6 64.4 48.4 35.7 28.9  24.5
tmax  11.3  13.0 17.5 24.0 27.4 31.1 33.2 32.7 27.8 22.9 15.8  12.0
tmin -14.2 -10.9 -6.0 -2.5  1.5  5.8  7.8  7.0  2.6 -2.3 -5.7 -10.7
tavg   0.2   1.2  4.3  9.2 13.9 17.3 19.5 19.2 14.5  9.7  4.8   1.4
```



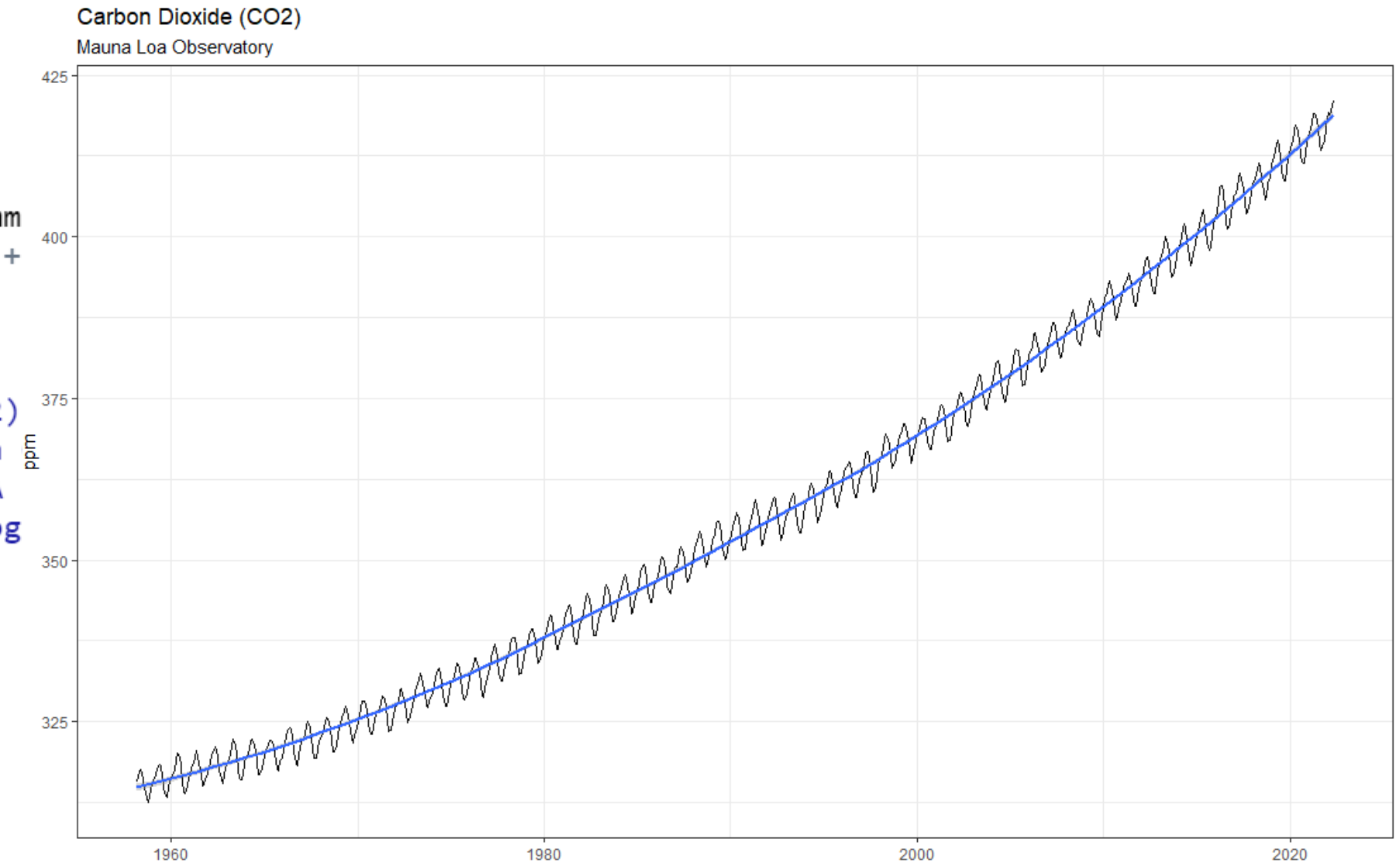
Przykład 3 – mapa trendów

```
library(ggplot2)
library(dplyr)
library(tidyr)
library(sf)
library(tmap)
library(rnaturalearth)
library(climate)
ms <- meteo_imgw("monthly", "synop", year = 1978:2017,
# calculating annual values
ms %>%
  filter(!(mm > 5 && mm < 9 && t2m_mean_mon == 0)) %>%
  select(station, X, Y, yy, mm, t2m_mean_mon) %>%
  group_by(station, yy, X, Y) %>%
  summarise(annual_mean_t2m = mean(t2m_mean_mon), n = n
  filter(n == 12) %>%
  spread(yy, annual_mean_t2m) %>%
  na.omit() -> trend
# extracting trends
regression <- function(x) {
df <- data.frame(yy = 1978:2017, temp = as.numeric(x))
      coef(lm(temp ~ yy, data = df))[2]
      }
trend$coef <- round(apply(trend[, -1:-4], 1, regression
trend <- st_as_sf(trend, coords = c("X", "Y"), crs = 43
# mapping the results
world <- ne_countries(scale = "medium", returnclass = "
tm <- tm_shape(world) + tm_borders() +
  tm_shape(trend, is.master = TRUE) + tm_dots(col = "co
  tm_shape(trend) + tm_text(text = "coef")+tmap_options
tm
```



Przykład 4 – stężenie CO_2

```
library(ggplot2)
library(ggthemes)

co2 = meteo_noaa_co2()
head(co2)
co2$date = ISOdate(co2$yy, co2$mm
ggplot(co2, aes(date, co2_avg)) +
  geom_line()+ geom_smooth()+
  theme_bw()+
  labs(
    title = "Carbon Dioxide (CO2)
    subtitle = paste0("Mauna Loa
    caption = "data source: NOAA
    visualization: Arkadiusz Głog
    x = "",
    y = "ppm"
  )
```



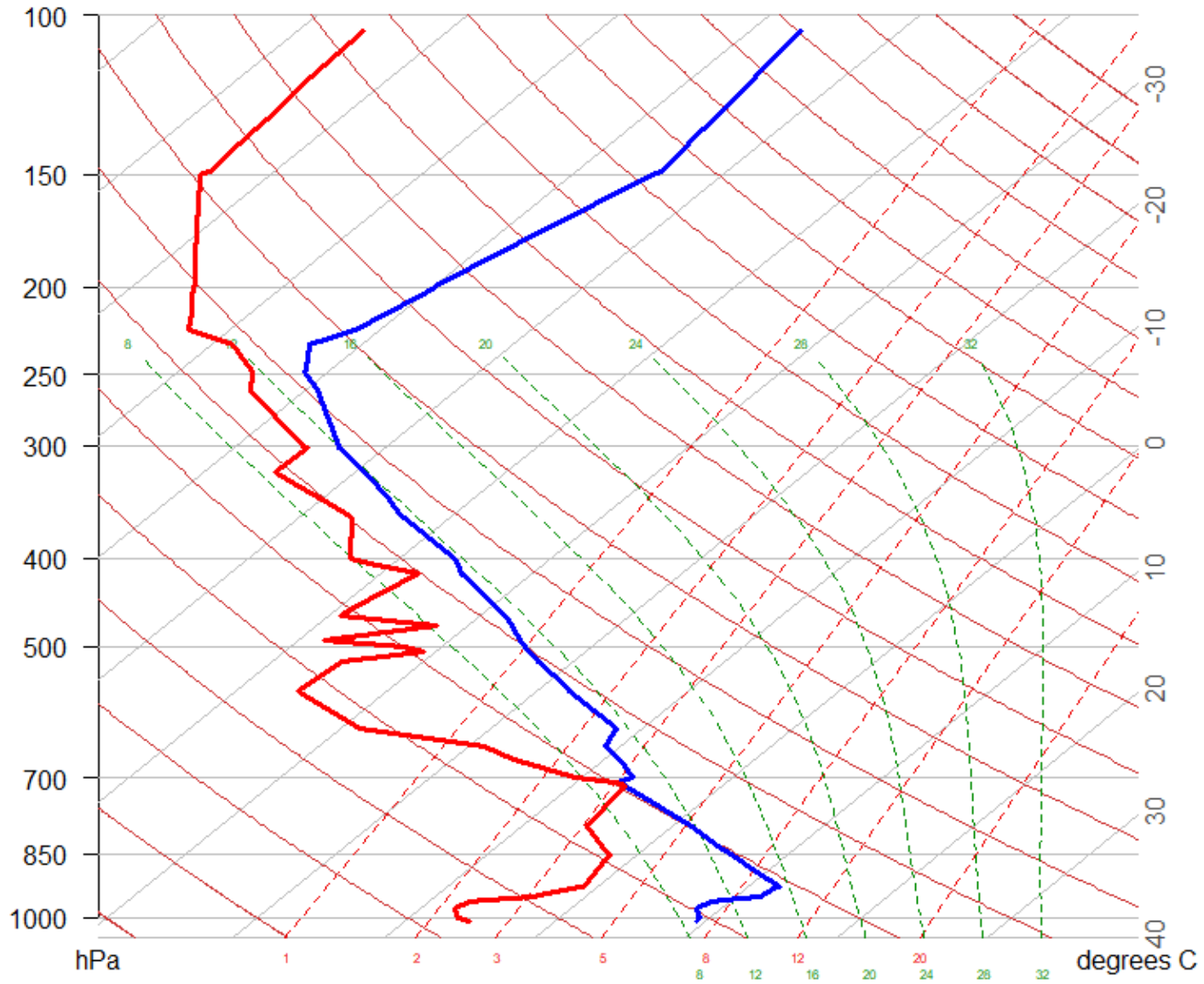
Dane radiosondażowe

- Wspierane zarówno depesze TEMP jak BUFR:

```
> ################################################################################
> # download data for Station 45004 starting 1120Z 11 Jul 2021; Kowloon, HONG KONG, CHINA
> # using TEMP and BUFR sounding formats
> ################################################################################
> TEMP = sounding_wyoming(wmo_id = 45004, yy = 2021, mm = 07, dd = 17, hh = 12, min = 00)
[1] "http://weather.uwyo.edu/cgi-bin/sounding?TYPE=TEXT%3ALIST&YEAR=2021&MONTH=07&FROM=1712&TO=1712&STNM=45004"
 [100%] Downloaded 10538 bytes...
/var/folders/lw/3fnnk87n4q1dkl35s2_p24h80000gn/T//RtmpRv7U4N/file79a33f07f3c
> head(TEMP[[1]])
  PRES HGHT TEMP DWPT RELH  MIXR DRCT SKNT  THTA  THTE  THTV
1 1000   50   NA   NA   NA    NA   NA   NA    NA    NA    NA
2  998   66 27.6 23.6   79 18.78    0    0 300.9 356.4 304.3
3  975  272 25.9 22.8   83 18.34  105   10 301.3 355.5 304.6
4  925  737 22.2 21.1   93 17.36  120   19 302.0 353.4 305.1
5  921  775 22.1 20.9   93 17.22  120   21 302.2 353.3 305.4
6  907  909 21.6 20.2   92 16.73  114   20 303.1 352.9 306.1
> BUFR = sounding_wyoming(wmo_id = 45004, yy = 2021, mm = 07, dd = 17, hh = 12, min = 00, bufr = TRUE)
[1] "http://weather.uwyo.edu/cgi-bin/bufrraob.py?datetime=2021-07-17+12:00:00&id=45004&type=TEXT:LIST"
 Downloaded 180521 bytes.../var/folders/lw/3fnnk87n4q1dkl35s2_p24h80000gn/T//RtmpRv7U4N/file79a70aa2082
> head(BUFR[[1]])
   PRES HGHT TEMP DWPT RELH  MIXR DRCT SKNT  THTA  THTE  THTV
1 998.2   66 27.7 23.7   79 18.85   27  0.4 301.0 356.7 304.4
2 996.1   84 27.7 23.8   79 19.00  120  4.7 301.2 357.4 304.6
3 995.1   93 27.7 23.8   79 18.96  120  5.6 301.3 357.4 304.7
4 993.9  104 27.6 23.6   79 18.76  120  6.0 301.3 356.8 304.7
5 992.7  115 27.5 23.5   79 18.69  120  6.4 301.3 356.6 304.6
6 991.5  125 27.4 23.5   79 18.67  120  6.8 301.2 356.5 304.6
```

Przykład 5 – Skew-T diagram

```
library(RadioSonde) # an external package
profile <- sounding_wyoming(wmo_id = 12120,y
df <- profile[[1]]
colnames(df)[c(1, 3:4)] = c("press", "temp",
RadioSonde::skewtPlot(df,colTemp = "blue",co
                      title = "2019-04-04 06
```



DZIĘKUJĘ ZA UWAGĘ !!!!!

UNIWERSYTET PRZYRODNICZY WE WROCŁAWIU

Kontakt:
arkadiusz.glogowski@upwr.edu.pl