API rNOMADS para recuperar dados de modelos climáticos NOAA

Previsão da precipitação acumulada (mm) para os próximos 15 dias

Vinicio Coelho Lima

Novembro, 2024

Contents

rNOMADS

NOMADS (NOAA Operational Model Archive and Distribution System) é um sistema de distribuição e arquivamento de dados operacionais gerenciado pela NOAA (National Oceanic and Atmospheric Administration). Ele fornece acesso a uma ampla gama de dados meteorológicos, climáticos e oceanográficos gerados por modelos numéricos de previsão e reanálises. Fonte: https://nomads.ncep.noaa.gov/.

O rNOMADS é uma interface para o sistema NOMADS que pode recuperar dados binários em formato grib, bem como importar dados ascii diretamente para o R por meio da interface com o sistema GrADS-DODS. Fonte: https://r-forge.r-project.org/projects/rnomads/.

```
library(rNOMADS)
# update.packages(oldPkgs = "rNOMADS")
library(raster)
```

Modelos

```
#  ascii
model.list  <-  NOMADSRealTimeList ( "dods" )

modelos <- aggregate(url ~ name,
                     data = model.list,
                     FUN = paste, collapse = ", ")

knitr::kable(modelos, row.names = TRUE)
```

	name	url
1	CMC Ensemble	https://nomads.ncep.noaa.gov:443/dods/cmcens/
2	FNMOC Ensemble and Bias Corrected	https://nomads.ncep.noaa.gov:443/dods/fens/
3	GDAS	https://nomads.ncep.noaa.gov:443/dods/fnl/
4	GDAS 0.25	https://nomads.ncep.noaa.gov:443/dods/gdas_0p25/
5	GFS 0.25 Degree	https://nomads.ncep.noaa.gov:443/dods/gfs_0p25/
6	GFS 0.25 Degree Hourly	https://nomads.ncep.noaa.gov:443/dods/gfs_0p25_1hr/
7	GFS 0.50 Degree	https://nomads.ncep.noaa.gov:443/dods/gfs_0p50/
8	GFS 1.00 Degree	https://nomads.ncep.noaa.gov:443/dods/gfs_1p00/
9	GFS Ensemble 0.5 Degree	https://nomads.ncep.noaa.gov:443/dods/gefs/
10	GFS Ensemble 0.5 Degree (Secondary Params)	https://nomads.ncep.noaa.gov:443/dods/gefs/
11	GFS Ensemble 0.5 Degree Bias-Corrected	https://nomads.ncep.noaa.gov:443/dods/gens_bc/
12	GFS Ensemble NDGD resolution Bias-Corrected	https://nomads.ncep.noaa.gov:443/dods/gens_ndgd/
13	GFS Wave	https://nomads.ncep.noaa.gov:443/dods/gfswave/
14	Great Lakes Wave Unstructured (GLWU)	https://nomads.ncep.noaa.gov:443/dods/glwu/
15	HIRESW Alaska	https://nomads.ncep.noaa.gov:443/dods/hiresw/
16	HIRESW CONUS	https://nomads.ncep.noaa.gov:443/dods/hiresw/
17	HIRESW Guam	https://nomads.ncep.noaa.gov:443/dods/hiresw/
18	HIRESW Hawaii	https://nomads.ncep.noaa.gov:443/dods/hiresw/
19	HIRESW Puerto Rico	https://nomads.ncep.noaa.gov:443/dods/hiresw/
20	HRRR	https://nomads.ncep.noaa.gov:443/dods/hrrr/
21	HRRR AK	https://nomads.ncep.noaa.gov:443/dods/hrrr/
22	NAEFS NDGD resolution Bias-Corrected	https://nomads.ncep.noaa.gov:443/dods/naefs_ndgd/
23	NAEFS high resolution Bias-Corrected	https://nomads.ncep.noaa.gov:443/dods/naefs_bc/
24	NAM Alaska Pressure Level Vars (11.25km)	https://nomads.ncep.noaa.gov:443/dods/nam/
25	NAM CONUS (12km)	https://nomads.ncep.noaa.gov:443/dods/nam/
26	NAM Caribbean/Central America	https://nomads.ncep.noaa.gov:443/dods/nam/
27	NAM NEST Alaska	https://nomads.ncep.noaa.gov:443/dods/nam/
28	NAM NEST CONUS	https://nomads.ncep.noaa.gov:443/dods/nam/
29	NAM NEST HAWAII	https://nomads.ncep.noaa.gov:443/dods/nam/
30	NAM NEST Puerto Rico	https://nomads.ncep.noaa.gov:443/dods/nam/
31	NAM North America (32km)	https://nomads.ncep.noaa.gov:443/dods/nam/
32	NAM Pacific	https://nomads.ncep.noaa.gov:443/dods/nam/
33	NARRE	https://nomads.ncep.noaa.gov:443/dods/narre/
34	NCEP and FNMOC Combined Ensemble Wave	https://nomads.ncep.noaa.gov:443/dods/nfcens/
35	National Blend of Models	https://nomads.ncep.noaa.gov:443/dods/blend/
36	RAP	https://nomads.ncep.noaa.gov:443/dods/rap/
37	RAP 32km North America	https://nomads.ncep.noaa.gov:443/dods/rap/
38	RAP Eastern North Pacific	https://nomads.ncep.noaa.gov:443/dods/rap/
39	RTMA ALASKA	https://nomads.ncep.noaa.gov:443/dods/akrtma/
40	RTMA Guam	https://nomads.ncep.noaa.gov:443/dods/gurtma/
41	RTMA Hawaii	https://nomads.ncep.noaa.gov:443/dods/hirtma/
42	RTMA Puerto Rico	https://nomads.ncep.noaa.gov:443/dods/prrtma/
43	RTMA2.5 CONUS	https://nomads.ncep.noaa.gov:443/dods/rtma2p5/
44	RTOFS Global	https://nomads.ncep.noaa.gov:443/dods/rtofs/
45	SREF CONUS (40km)	https://nomads.ncep.noaa.gov:443/dods/sref/
46	SREF CONUS (40km) Bias-Corrected	https://nomads.ncep.noaa.gov:443/dods/sref_bc/
47	SREF North America (16km)	https://nomads.ncep.noaa.gov:443/dods/sref/

	name	url
48	SREF North America (32km)	https://nomads.ncep.noaa.gov:443/dods/sref/
49	STOFS 2D Global	https://nomads.ncep.noaa.gov:443/dods/stofs_2d_glo/
50	STOFS 3D Atlantic	https://nomads.ncep.noaa.gov:443/dods/stofs_3d_atl/
51	Sea Ice Analysis	https://nomads.ncep.noaa.gov:443/dods/ice/

Global Forecast System

O Global Forecast System (GFS) é um modelo de previsão do tempo produzido pelo National Centers for Environmental Prediction (NCEP). Dezenas de variáveis atmosféricas e de solo estão disponíveis através deste conjunto de dados, desde temperaturas, ventos e precipitação até umidade do solo e concentração de ozônio na atmosfera.

Mudanças são feitas regularmente no modelo GFS para melhorar seu desempenho e precisão de previsão. Este conjunto de dados é executado quatro vezes ao dia às 00z, 06z, 12z e 18z até 192 horas com uma resolução horizontal de 0,5 graus e uma resolução temporal de 3 horas.

gfs_0p50

O modelo "0p50" refere-se a uma grade global com intervalos de 0,50 graus (~55 km entre os pontos de grade na linha do Equador). Oferece um nível intermediário de detalhamento espacial em comparação a versões mais finas (0,25 graus) ou mais grossas (1,00 grau).

Fornece previsões de curto a médio prazo (até 384 horas, ou 16 dias). As saídas podem ser em intervalos de 3 ou 6 horas, dependendo da configuração.

```
model.urls <- GetDODSDates("gfs_0p50")
```

Região de interesse

```
lat <- -19.78753
lon <- -51.98899
```

Configurando a grade

```
# Subset
lons <- seq(0, 359.5, by = 0.5)
lats <- seq(-90, 90, by = 0.5)

lon.diff <- abs(lon + 360 - lons)
lat.diff <- abs(lat - lats)

model.lon.ind <- which(lon.diff == min(lon.diff)) - 1 # Indexado no 0
model.lat.ind <- which(lat.diff == min(lat.diff)) - 1


lon.inds <- c(model.lon.ind - 12, model.lon.ind + 12) # região
lat.inds <- c(model.lat.ind - 14, model.lat.ind + 14)
```

Modelo mais recente

```
latest.model <- tail(model.urls$url, 1)
model.runs <- GetDODSModelRuns(latest.model)
model.runs
```

```
## $model.run
## [1] "gfs_0p50_00z" "gfs_0p50_06z" "gfs_0p50_12z"
##
## $model.run.info
## [1] "gfs_0p50_00z: GFS 0.5 deg starting from 00Z17nov2024, downloaded Nov 17 05:15 UTC"
## [2] "gfs_0p50_06z: GFS 0.5 deg starting from 06Z17nov2024, downloaded Nov 17 11:11 UTC"
## [3] "gfs_0p50_12z: GFS 0.5 deg starting from 12Z17nov2024, downloaded Nov 17 17:13 UTC"
```

```
latest.model.run <- tail(model.runs$model.run, 1)
latest.model.run
```

```
## [1] "gfs_0p50_12z"
```

Variáveis disponíveis

```
model.info <- GetDODSModelRunInfo(latest.model, tail(model.runs$model.run, 1))
model.info.var <- model.info[c(10,11,28,43,141:148,170,180,201:204,216,220,239)]

knitr::kable(as.data.frame(model.info.var), row.names = TRUE)
```

	model.info.var
1	acpcpsfc ** surface convective precipitation [kg/m^2]
2	albdosfc ** surface albedo [%]
3	cnwatsfc ** surface plant canopy surface water [kg/m^2]
4	fldcpsfc ** surface field capacity [fraction]
5	soill0_10cm ** 0-0.1 m below ground liquid volumetric soil moisture (non frozen) [proportion]
6	soill10_40cm ** 0.1-0.4 m below ground liquid volumetric soil moisture (non frozen) [proportion]
7	soill40_100cm ** 0.4-1 m below ground liquid volumetric soil moisture (non frozen) [proportion]
8	soill100_200cm ** 1-2 m below ground liquid volumetric soil moisture (non frozen) [proportion]
9	soilw0_10cm ** 0-0.1 m below ground volumetric soil moisture content [fraction]
10	soilw10_40cm ** 0.1-0.4 m below ground volumetric soil moisture content [fraction]
11	soilw40_100cm ** 0.4-1 m below ground volumetric soil moisture content [fraction]
12	soilw100_200cm ** 1-2 m below ground volumetric soil moisture content [fraction]
13	tmpsfc ** surface temperature [k]
14	tmp2m ** 2 m above ground temperature [k]
15	tsoil0_10cm ** 0-0.1 m below ground soil temperature validation to deprecate [k]
16	tsoil10_40cm ** 0.1-0.4 m below ground soil temperature validation to deprecate [k]
17	tsoil40_100cm ** 0.4-1 m below ground soil temperature validation to deprecate [k]
18	tsoil100_200cm ** 1-2 m below ground soil temperature validation to deprecate [k]
19	ugrd10m ** 10 m above ground u-component of wind [m/s]
20	ugrd50m ** 50 m above ground u-component of wind [m/s]
21	ugrdmwl ** max wind u-component of wind [m/s]

Precipitação

```
variables <- "acpcpsfc" # Accumulated precipitation surface (mm)
```

Previsão

A variável time no DODSGrab() é um vetor de dois componentes, como c(start, end), que define os índices do intervalo de tempo. O modelo GFS fornece previsões a cada 3 horas:

- time = c(0, 0) Previsão para o tempo mais atual.
- time = c(8, 8) 24 horas à frente (8 × 3 horas = 24 horas).
- time = c(116, 116) 116 × 3 horas = 348 horas, ou cerca de 14,5 dias.

```
time <- c(116,116) # Status de inicialização

model.data <- DODSGrab(latest.model, latest.model.run, variables,
                       time, lon.inds , lat.inds)
```

```
## [1] "https://nomads.ncep.noaa.gov:443/dods/gfs_0p50/gfs20241117/gfs_0p50_12z.ascii?acpcpsfc[116:116]
```

```
hoje <- Sys.time()
hoje
```

```
## [1] "2024-11-17 15:34:44 -03"
```

```
forecast <- ModelGrid(model.data, c(0.5, 0.5))
forecast$fcst.date
```

```
## [1] "2024-12-02 GMT"
```

Mapa para o modelo

```
prec <- list()
prec$x <- forecast$x
prec$y <- forecast$y
prec$z <- forecast$z[1,1,,]
r <- raster::raster(prec)
r2 <- raster::rotate(r)
crs_target <- CRS("+proj=longlat +datum=WGS84 +no_defs")
r3 <- projectRaster(r2, crs = crs_target)
# Converter raster para data.frame para ggplot
r_df <- as.data.frame(r3, xy = TRUE, na.rm = TRUE)
colnames(r_df) <- c("lon", "lat", "precip")
```

Base Cartográfica IBGE

```
library(dplyr)
library(sf)

# xmin       ymin       xmax       ymax # wkt filter - limite da consulta .gpkg
# -61.633383 -33.751178 -39.856829  -7.349028

pts <- st_sf(pt = 1:2,
             geom = st_sfc(st_point(c(
               -61.633383, -33.751178
             )), st_point(c(
               -39.856829, -7.349028
             ))),
             crs = 4326)
pol <- pts %>% st_bbox() %>% st_as_sfc(., crs = 4326)
```

```r
subset_reg <- st_as_text(st_geometry(pol))

bc_ibge <- "/IBGE/BC_250/bc250_ibge.gpkg"
st_layers(bc_ibge)

# Sys.setlocale("LC_ALL", "pt_BR.UTF-8")
cidade <- st_read(bc_ibge, layer = "lml_cidade_p",
                  wkt_filter = subset_reg) %>% # Evita carregar todos o municipios
  dplyr::filter(nome %in% c("Bataguassu",#MS
                            "Inocência", #MS
                            "Dourados", # MS
                            "Ortigueira", #PR
                            "Cascavel", # PR
                            "Uberlândia", # MG
                            "Bauru", # SP
                            "Ribeirão Preto", # SP
                            "Rio Verde", # GO
                            "Alto Araguaia", # GO
                            "Passo Fundo")) # RS

capital <- st_read(bc_ibge, layer = "lml_capital_p") %>% dplyr::select( - tipocapital)
mun_label <- rbind(cidade, capital)

uf <- st_read(bc_ibge, layer = "lml_unidade_federacao_a")

mun_label_coords <- cbind(mun_label, st_coordinates(mun_label))
```

Plot

```r
library(ggplot2)

ext.mapa <- c(xmin = -58.0, xmax = -46.25, ymin = -28, ymax = -13)

# Definir a codificação de caracteres para UTF-8
# Sys.setlocale("LC_ALL", "pt_BR.UTF-8")

 p <- ggplot() +
   geom_raster(data = r_df, aes(x = lon, y = lat, fill = precip)) +
   scale_fill_gradientn(
     colors = c("#FFFFCC", "#41B6C4", "#0C2C84"),
     name = "",
     na.value = "transparent"
   ) +
   geom_sf(
     data = uf,
     fill = "lightblue",
     color = "black",
     alpha = .10
   ) +
   geom_sf(data = mun_label,
           color = "red",
           size = .15) +
   geom_text(
```

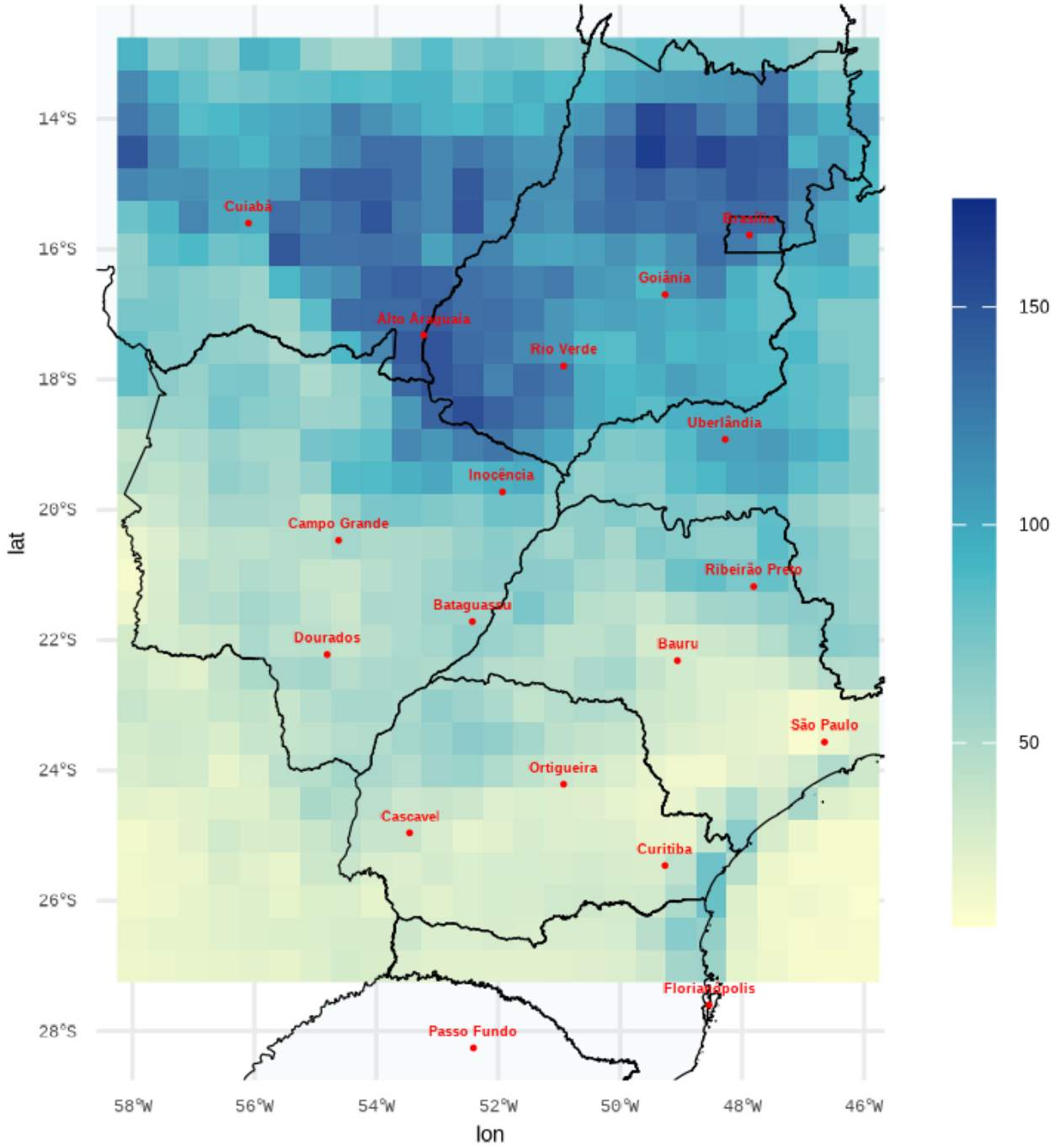
```
    data = mun_label_coords,
    aes(x = X, y = Y, label = nome),
    size = 2.25,
    nudge_y = .25,
    color = "red",
    fontface = "bold",
    check_overlap = TRUE
  ) +
  coord_sf(xlim = c(ext.mapa["xmin"], ext.mapa["xmax"]),
           ylim = c(ext.mapa["ymin"], ext.mapa["ymax"])) +
  theme_minimal()  +
  labs(
    title = "Modelo GFS 0.5 deg NCEP",
    subtitle = "Precipitação acumulada (mm) em 348 horas
             <2024-12-02 00:00:00 GMT>",
    caption = "gfs_0p50_12z: GFS 0.5 deg starting from 12Z17nov2024, 17:13 UTC"
  ) +
  theme(
    plot.title = element_text(
      size = 22,
      face = "bold",
      family = "serif"
    ),
    plot.subtitle = element_text(
      size = 12,
      face = "plain",
      family = "mono"
    ),
    plot.caption = element_text(
      size = 10,
      face = "plain",
      family = "mono"
    ),
    axis.text = element_text(family = "mono"),
    axis.text.x = element_text(family = "mono"),
    axis.text.y = element_text(family = "mono"),
    legend.key.height = unit(0.10, 'npc'),
    legend.key.width = unit(0.04, 'npc')
  )

ggsave(
  plot = p,
  filename = "./mapa_modelo_gfs.png",
  width = 9.5,
  height = 12.5,
  units = "cm",
  device = "png",
  dpi = 200,
  bg = "white"
)
```

Modelo GFS 0.5 deg NCEP

Precipitação acumulada (mm) em 348 horas

<2024-12-02 00:00:00 GMT>



gfs_0p50_12z: GFS 0.5 deg starting from 12Z17nov2024, 17:13 UTC