OTNuEC

January 8, 2026

1 OTNuEC

2 1) Environment

[1]:
```r
ROOT <- "C:/Users/Pedro/Desktop/Dados OTNuEC"
p <- function(...) file.path(...)

DIRS <- list(
  bc250   = p(ROOT, "bc250 2025"),
  erosao  = p(ROOT, "Chance de Erosao"),
  dnit    = p(ROOT, "Dados Dnit"),
  mapbio  = p(ROOT, "Mapbiomas"),
  solos   = p(ROOT, "Qualidade do Solo"),
  rais    = p(ROOT, "RAIS"),
  etopo   = p(ROOT, "Relevo ETOPO"),
  viirs   = p(ROOT, "VIIRS"),
  sidra   = p(ROOT, "SIDRA (Faltando)")
)

# ---------------- BC250 ----------------
BC250_SHAPE <- p(DIRS$bc250, "shapefile")

bc250 <- list(
  roads   = p(BC250_SHAPE, "rod_trecho_rodoviario_l.shp"),
  rail    = p(BC250_SHAPE, "fer_trecho_ferroviario_l.shp"),
  hydro   = p(BC250_SHAPE, "hdv_trecho_hidroviario_l.shp"),
  ports   = p(BC250_SHAPE, "hdv_complexo_portuario_p.shp"),
  locks   = p(BC250_SHAPE, "hdv_eclusa_p.shp"),
  bridges = p(BC250_SHAPE, "tra_ponte_l.shp"),
  tunnels = p(BC250_SHAPE, "tra_tunel_l.shp"),
  ferries = p(BC250_SHAPE, "tra_travessia_l.shp"),
  power   = p(BC250_SHAPE, "enc_trecho_energia_l.shp"),
  hydro_p = p(BC250_SHAPE, "enc_hidreletrica_p.shp"),
  thermo  = p(BC250_SHAPE, "enc_termeletrica_p.shp"),
  subs    = p(BC250_SHAPE, "enc_subest_transm_distrib_energia_eletrica_p.shp"),
  dams    = p(BC250_SHAPE, "hid_barragem_p.shp"),
  airports= p(BC250_SHAPE, "aer_complexo_aeroportuario_p.shp")
```

```
)

# ---------------- DNIT ----------------
dnit <- DIRS$dnit

DNIT <- list(
  pav = list(
    pav = p(dnit, "Condicoes do Pavimento", "levantamentos_pavimentada_2025_10.
 ↪csv"),
    nao = p(dnit, "Condicoes do Pavimento",␣
 ↪"levantamentos_nao_pavimentada_2025_10.csv")
  ),
  od = list(
  postos   = p(dnit, "Pesquisa OD", "Postos PNT 2016-2017"),
  matrizes = p(dnit, "Pesquisa OD", "Matrizes PNT 2016-2017.xlsx"),
  micro    = p(dnit, "Pesquisa OD", "PesquisaOD 2016-2017.xlsx"),
  nota     = p(dnit, "Pesquisa OD", "NotaTécnica.pdf")
),

vmda = list(
  tabela = p(dnit, "VMDa", "VMDa 2023.xlsx"),
  nota   = p(dnit, "VMDa", "NotaTécnica VMDA 2023.pdf"),
  geo    = p(dnit, "VMDa", "(Camada Geo) VMDA - Modelagem 2023")
)
)
# ---------------- Rasters & tables ----------------
DATA <- list(
  root = ROOT,
  bc250 = bc250,
  erosao = list(raster = p(DIRS$erosao, "BR_vulnerabilidade_solos_erosao_2020.
 ↪tif")),
  etopo  = list(raster = p(DIRS$etopo, "ETOPO1_Ice_g_geotiff.tif")),
  viirs  = list(blackmarble_2016 = p(DIRS$viirs, "viirs_blackmarble_br_2016.
 ↪tif")),
  mapbiomas = list(forest_2024 = p(DIRS$mapbio, "brazil_coverage_2024.tif")),
  solos = list(wrb = p(DIRS$solos, "soils_brazil_wrb_wgs84.shp")),
  rais  = list(empregos = p(DIRS$rais, "consulta9293258.csv")),
  dnit  = DNIT
)
```

[2]:
```
check_path <- function(x) {
  ok <- file.exists(x)
  if (!ok) message("FALTA: ", x)
  ok
}

check_all <- function(lst) {
```

```
  unlist(lapply(lst, function(x) {
    if (is.list(x)) check_all(x) else check_path(x)
  }))
}

check_all(DATA)
```

root TRUE **bc250.roads** TRUE **bc250.rail** TRUE **bc250.hydro** TRUE **bc250.ports** TRUE **bc250.locks** TRUE **bc250.bridges** TRUE **bc250.tunnels** TRUE **bc250.ferries** TRUE **bc250.power** TRUE **bc250.hydro_p** TRUE **bc250.thermo** TRUE **bc250.subs** TRUE **bc250.dams** TRUE **bc250.airports** TRUE **erosao.raster** TRUE **etopo.raster** TRUE **viirs.blackmarble_2016** TRUE **mapbiomas.forest_2024** TRUE **solos.wrb** TRUE **rais.empregos** TRUE **dnit.pav.pav** TRUE **dnit.pav.nao** TRUE **dnit.od.postos** TRUE **dnit.od.matrizes** TRUE **dnit.od.micro** TRUE **dnit.od.nota** TRUE **dnit.vmda.tabela** TRUE **dnit.vmda.nota** TRUE **dnit.vmda.geo** TRUE

[3]:
```
library(sf)
library(geobr)
library(terra)
library(dplyr)
```

```
Warning message:
"pacote 'sf' foi compilado no R versão 4.5.2"
Linking to GEOS 3.13.1, GDAL 3.11.4, PROJ 9.7.0; sf_use_s2() is TRUE

Warning message:
"pacote 'geobr' foi compilado no R versão 4.5.2"
Warning message:
"pacote 'terra' foi compilado no R versão 4.5.2"
terra 1.8.80


Anexando pacote: 'dplyr'


Os seguintes objetos são mascarados por 'package:terra':

    intersect, union


Os seguintes objetos são mascarados por 'package:stats':

    filter, lag


Os seguintes objetos são mascarados por 'package:base':

    intersect, setdiff, setequal, union
```

[4]:
```r
AUX <- file.path(DATA$root, "aux_data")
dir.create(AUX, showWarnings = FALSE, recursive = TRUE)
NODES_FILE <- file.path(AUX, "nodes_micro_viirs.gpkg")
```

3 2) Data Display

3.1 2.1) Nodes Position

[5]:
```r
# Microrregiões oficiais do IBGE
micro <- read_micro_region(year = 2020, simplified = FALSE) |>
  st_make_valid() |>
  st_transform(5880)   # SIRGAS / Brasil métrico
```

```
Using year/date 2020
```

[6]:
```r
bbox_mainland <- st_bbox(c(
  xmin = -75, xmax = -34,
  ymin = -34, ymax =  6
), crs = 4326) |> st_as_sfc() |> st_transform(5880)

micro <- st_intersection(micro, bbox_mainland)
```

```
Warning message:
"attribute variables are assumed to be spatially constant throughout all
geometries"
```

[7]:
```r
if (file.exists(NODES_FILE)) {

  message("→ carregando nodes_micro do disco")
  nodes_micro <- st_read(NODES_FILE, quiet = TRUE)

} else {

  message("→ construindo nodes_micro a partir do VIIRS")

  viirs <- rast(DATA$viirs$blackmarble_2016) |>
    project("EPSG:5880")

  ntl <- terra::rast(DATA$viirs$blackmarble_2016)
  names(ntl) <- "value"

  build_nodes_simple <- function(regions_sf, ntl_rast) {
    reg <- sf::st_transform(regions_sf, terra::crs(ntl_rast))
```

```
    vals <- terra::extract(ntl_rast, terra::vect(reg), cells = TRUE)

    best <- vals |>
      group_by(ID) |>
      filter(!is.na(value)) |>
      slice_max(order_by = value, n = 1, with_ties = FALSE) |>
      ungroup()

    xy <- terra::xyFromCell(ntl_rast, best$cell)

    sf::st_as_sf(
      data.frame(
        code_micro = reg$code_micro[best$ID],
        lon = xy[, 1],
        lat = xy[, 2]
      ),
      coords = c("lon", "lat"),
      crs = terra::crs(ntl_rast)
    )
  }

  nodes_micro <- build_nodes_simple(micro, ntl)

  st_write(nodes_micro, NODES_FILE, delete_dsn = TRUE)
}
```

```
→ carregando nodes_micro do disco
```

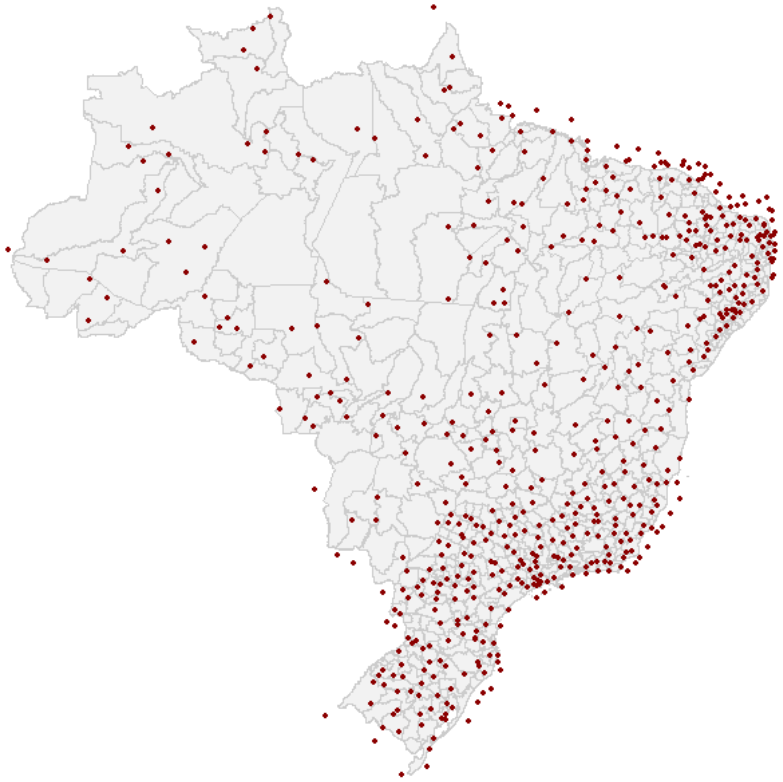
[18]:
```
viirs <- rast(DATA$viirs$blackmarble_2016) |>
    project("EPSG:5880")

ntl <- terra::rast(DATA$viirs$blackmarble_2016)
  names(ntl) <- "value"
```

[19]:
```
build_nodes_simple <- function(regions_sf, ntl_rast) {

  # garantir mesmo CRS
  regions_ll <- st_transform(regions_sf, st_crs(ntl_rast))

  # extrair valores do raster dentro de cada polígono
  vals <- terra::extract(ntl_rast, terra::vect(regions_ll), cells = TRUE)

  best <- vals |>
    dplyr::group_by(ID) |>
    dplyr::filter(!is.na(value)) |>
    dplyr::slice_max(order_by = value, n = 1, with_ties = FALSE) |>
    dplyr::ungroup()
```

```
  xy <- terra::xyFromCell(ntl_rast, best$cell)

  sf::st_as_sf(
    data.frame(
      id = regions_ll$code_micro[best$ID],
      lon = xy[,1],
      lat = xy[,2]
    ),
    coords = c("lon", "lat"),
    crs = st_crs(ntl_rast)
  )
}
  nodes_micro <- build_nodes_simple(micro, ntl)

  st_write(nodes_micro, NODES_FILE, delete_dsn = TRUE)
```

```
Deleting source `C:/Users/Pedro/Desktop/Dados
OTNuEC/aux_data/nodes_micro_viirs.gpkg' using driver `GPKG'
Writing layer `nodes_micro_viirs' to data source
  `C:/Users/Pedro/Desktop/Dados OTNuEC/aux_data/nodes_micro_viirs.gpkg' using
driver `GPKG'
Writing 557 features with 1 fields and geometry type Point.
```

[20]:
```
plot(st_geometry(micro),
     col = "grey95", border = "grey80", lwd = 0.5)

par(new = TRUE)
plot(st_geometry(nodes_micro),
     pch = 20, col = "darkred", cex = 0.7,
     axes = FALSE, xlab = "", ylab = "")
```



4 2.2) Protected Areas

[ ]:
```
if (file.exists("aux_data/pa_uc_ti.rds")) {
  pa_uc_ti <- readRDS("aux_data/pa_uc_ti.rds")
} else {
uc <- geobr::read_conservation_units()
ti <- geobr::read_indigenous_land()

bind_rows(
  uc |> mutate(src = "UC"),
  ti |> mutate(src = "TI")
)
```

```r
dir.create("aux_data", showWarnings = FALSE)

pa_uc_ti <- bind_rows(
  geobr::read_conservation_units() |> mutate(src = "UC"),
  geobr::read_indigenous_land()    |> mutate(src = "TI")
) |>
  st_make_valid()

saveRDS(pa_uc_ti, "aux_data/pa_uc_ti.rds")
}
```

[ ]: