WRF GeoGrid Support

Anamika Shreevastava

9/17/2019

This script helps in designing domain setups for WRF runs.

Functions

These functions are written for concentric square domains only for now.

```
# save api key
register_google(key = "AIzaSyBGc7jBM4D1uZigyrO1XVHWXhyrytdUG5E")

bounding_box <- function(lat, lon, dist, in.miles = FALSE) {
  ##Source: http://rstudio-pubs-static.s3.amazonaws.com/19324_dd865f50a2304595b45d86f3022f4681.html
  ## Helper functions
  if (in.miles) {
    ang_rad <- function(miles) miles/3958.756
  } else {
    ang_rad <- function(miles) miles/(1.60934*3958.756)
  }
  `%+/-%` <- function(x, margin){x + c(-1, +1)*margin}
  deg2rad <- function(x) x/(180/pi)
  rad2deg <- function(x) x*(180/pi)
  lat_range <- function(latr, r) rad2deg(latr %+/-% r)
  lon_range <- function(lonr, dlon) rad2deg(lonr %+/-% dlon)

  r <- ang_rad(dist)
  latr <- deg2rad(lat)
  lonr <- deg2rad(lon)
  dlon <- asin(sin(r)/cos(latr))

  m <- matrix(c(lon_range(lonr = lonr, dlon = dlon),
                lat_range(latr=latr, r=r)), nrow=2, byrow = TRUE)

  dimnames(m) <- list(c("lng", "lat"), c("min", "max"))
  m
}
Plot_Domains <- function(City, Dom1, Dom2, Dom3, zoom_level){

  CityLon <- geocode(City)[[1]]
  CityLat <- geocode(City)[[2]]

  bb1 <- bounding_box(lon = CityLon, lat = CityLat, dist=Dom1/2, in.miles=F)
  bb2 <- bounding_box(lon = CityLon, lat = CityLat, dist=Dom2/2, in.miles=F)
  bb3 <- bounding_box(lon = CityLon, lat = CityLat, dist=Dom3/2, in.miles=F)

  if (zoom_level==1) {
    box <- as.vector(bb1)
    names(box) <- c("left","bottom","right","top")
    zoom <- calc_zoom(box)
  } else if (zoom_level==2){
    box <- as.vector(bb2)
    names(box) <- c("left","bottom","right","top")
    zoom <- calc_zoom(box) - 1
  } else if (zoom_level==3){
    box <- as.vector(bb3)
    names(box) <- c("left","bottom","right","top")
    zoom <- calc_zoom(box) - 1
  } else {
    zoom <- zoom_level
  }
  print(paste0("Zoom Level is ",zoom))

  mapImageData <- get_map(location = c(lon = CityLon, lat = CityLat),
                  color = "color", source = "google", maptype = "roadmap",
                  zoom = zoom)
  ## zoom is an integer value from 3 (continent) to 21 (building)

  ggmap(mapImageData, extent= "panel", ylab = "Latitude", xlab = "Longitude") +
    geom_rect(aes(xmin = bb1[1, 1], xmax = bb1[1, 2], ymin = bb1[2, 1],
                  ymax = bb1[2, 2]), color="red", fill=NA, size=1) +
    geom_rect(aes(xmin = bb2[1, 1], xmax = bb2[1, 2], ymin = bb2[2, 1],
                  ymax = bb2[2, 2]), color="blue", fill=NA, size=1) +
    geom_rect(aes(xmin = bb3[1, 1], xmax = bb3[1, 2], ymin = bb3[2, 1],
                  ymax = bb3[2, 2]), color="green", fill=NA, size=1) +
    geom_point(data=data.frame(lon = CityLon, lat = CityLat), size=5,
               color="yellow")

}
WRF_domain <- function(City, Dom1, Dom2, Dom3){

  GridCells1 <- Dom1/9     ## GridCells in Domain 1 of size 9 km each
  GridCells1 <- 9*ceiling(GridCells1/9)   ## Rounding off to evenly fit

  GridCells2 <- Dom2/3     ## GridCells in Domain 2 of size 3 km each
  GridCells2<- 18*ceiling(GridCells2/18) ## Rounding off to evenly fit
  GridCells2_1 <- GridCells2/3   ## GridCells in Domain 2 of size 9 km each

  GridCells3 <- Dom3/1     ## GridCells in Domain 3 of size 1 km each
  GridCells3<- 18*ceiling(GridCells3/18) ## Rounding off to evenly fit
  GridCells3_1 <- GridCells3/9   ## GridCells in Domain 3 of size 9 km each

#  print(paste0("Number of Grid Cells in Domain 1 of size 9 km each = ", GridCells1))
#  print(paste0("Number of Grid Cells in Domain 2 of size 3 km each = ", GridCells2))
#  print(paste0("Number of Grid Cells in Domain 3 of size 1 km each = ", GridCells3))

  if(GridCells1 < 100) warning("The number of gridcells in Domain 1 should not be less than 100")
  if(GridCells2 < 100) warning("The number of gridcells in Domain 2 should not be less than 100")
  if(GridCells3 < 100) warning("The number of gridcells in Domain 3 should not be less than 100")

  if((GridCells1 - GridCells2_1)/2 < 10) warning("There should be at least 10 large gridcells in the bou
  if((GridCells2_1 - GridCells3_1)*3/2 < 10) warning("There should be at least 10 medium gridcells in th
  Start2 <- (GridCells1 - GridCells2_1)/2
  Start3 <- (GridCells1 - GridCells3_1)/2
#  e_we <- data.table(GridCells1, GridCells2+1,GridCells3+1)%>%as.data.frame()
#  i_parent_start <- data.table(1, Start2, Start3)%>%as.data.frame()
  e_we <- c(GridCells1, GridCells2+1,GridCells3+1)
  i_parent_start <- c(1, Start2, Start3)

  WRF_out <- data.table(rbind(c("parent_id",1,1,1),
                        c("parent_grid_ratio",1,3,9),
                        c("i_parent_start",i_parent_start),
                        c("j_parent_start",i_parent_start),
                        c("e_we",e_we),
                        c("e_sn",e_we)))

  colnames(WRF_out) <-  c("Label", "Domain 1", "Domain 2", "Domain 3")
  print(WRF_out)
  return(WRF_out)
}
```

Input City's name here and get Latitude Longitude of center of domains

```
City <- "Paris"

Ref_Lon <- geocode(City)[[1]]
```

```
## Source : https://maps.googleapis.com/maps/api/geocode/json?address=Paris&key=AIzaSyBGc7jBM4D1uZigyr0
```

```
Ref_Lat <- geocode(City)[[2]]
```

```
## Source : https://maps.googleapis.com/maps/api/geocode/json?address=Paris&key=AIzaSyBGc7jBM4D1uZigyr0
```

Get GeoGrid details

Enter the size of square domains you need (edge length in km)

```
Dom1 <- 929 # Size of domain 1
Dom2 <- 500 # Size of domain 2
Dom3 <- 200 # Size of domain 3

print(paste0("Ref_lat = ", Ref_Lat, "; Ref_lon = ", Ref_Lon))
```

```
## [1] "Ref_lat = 48.856614; Ref_lon = 2.3522219"
```

```
geogrid <- WRF_domain(City, Dom1, Dom2, Dom3)
```

```
##                Label Domain 1 Domain 2 Domain 3
## 1:         parent_id        1        1        1
## 2: parent_grid_ratio        1        3        9
## 3:    i_parent_start        1       24       42
## 4:    j_parent_start        1       24       42
## 5:              e_we      108      181      217
## 6:              e_sn      108      181      217
```

Visualize domains

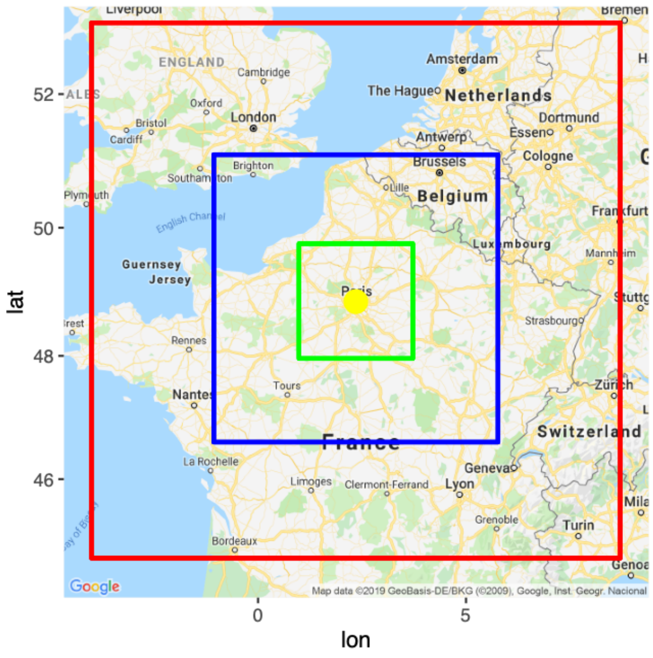
Set Zoom_level to whichever domain you want to focus on

```
Plot_Domains(City,Dom1,Dom2,Dom3,zoom_level= 1)
```

```
## Source : https://maps.googleapis.com/maps/api/geocode/json?address=Paris&key=AIzaSyBGc7jBM4D1uZigyr0
## Source : https://maps.googleapis.com/maps/api/geocode/json?address=Paris&key=AIzaSyBGc7jBM4D1uZigyr0
```

```
## [1] "Zoom Level is 6"
```

```
## Source : https://maps.googleapis.com/maps/api/staticmap?center=48.856614,2.352222&zoom=6&size=640x64
```



lat
lon
52
50
48
46
0
5
Liverpool
Bremen
ENGLAND
Cambridge
Amsterdam
The Hague
Netherlands
Oxford
London
Bristol
Cardiff
Dortmund
Essen
Antwerp
Cologne
Brussels
Brighton
Southampton
Lille
Belgium
Plymouth
English Channel
Frankfurt
Luxembourg
Mannheim
Guernsey
Jersey
Paris
Stuttg
Strasbourg
Rennes
Nantes
Tours
Zurich
Switzerland
France
La Rochelle
Geneva
Limoges
Clermont-Ferrand
Lyon
Milan
Grenoble
Turin
Bordeaux
Genoa
Google
Map data ©2019 GeoBasis-DE/BKG (©2009), Google, Inst. Geogr. Nacional