

# Programação em R para Geoprocessamento



Pedro José Farias Fernandes  
[pj\\_fernandes@id.uff.br](mailto:pj_fernandes@id.uff.br)  
Laboratório de Geografia Física

# Pacote sp

- Uma interface para manipular dados geoespaciais
- Ele cria novas classes de dados geoespaciais, e ferramentas para sua manipulação no R
- Essas classes servem para que outros pacotes as usem
- <https://cran.r-project.org/web/packages/sp/sp.pdf>

# Pacote rgdal

- Há funções para acessar arquivos raster e vetoriais.
- Mas há similares no pacote raster.
- <https://cran.r-project.org/web/packages/rgdal/rgdal.pdf>

# Leitura de shapefiles

```
install.packages("sp")
```

```
install.packages("raster")
```

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

```
library(sp)
```

```
library(rgdal)
```

```
niteroi<-readOGR("/home/uff/Área de  
Trabalho/arquivos/curso_r/aula_4/pratica_1_shapefile/Bairros_Niteroi.shp", "Bairros_Niteroi")
```

```
plot(niteroi)
```

```
summary(niteroi)
```

# Leitura de shapefiles

```
#Vamos fazer o subset do bairro Boa Viagem
```

```
bv<-niteroi[niteroi$NOME=="BOA VIAGEM",]
```

```
plot(bv)
```

```
crs(bv) #FUNÇÃO DO PACOTE RASTER (PROJEÇÃO)
```

```
#proj4string(bv)<-CRS() #APAGANDO A PROJEÇÃO
```

```
proj4string(bv)<-CRS("+proj=utm +zone=23 +south +datum=WGS84 +units=m +no_defs +ellps=WGS84 +towgs84=0,0,0") #DEFININDO A PROJEÇÃO
```

```
crs(bv)
```

```
bv_longlat<-spTransform(bv,CRS("+proj=longlat")) #REPROJENTADO PARA LONGLAT
```

```
#PLOT
```

```
plot(bv_longlat)
```

```
lgridlines(bv_longlat,lty=3) #PLOTAR O GRID. Lty é o tipo
```

```
writeOGR(bv_longlat,"LL","bv_ll",driver="ESRI Shapefile") #no pacote rgdal
```

```
shapefile(bv,"bv_ll.shp") #no pacote raster
```

# Lendo dados de DGPS



# Lendo dados de DGPS

```
install.packages("gstat")
```

```
library(gstat)
```

```
library(raster)
```

```
library(rgdal)
```

```
library(sp)
```

```
pontos<-read.csv("https://raw.githubusercontent.com/pjfernandes/curso_r/master/aula_4/pratica_2_leitura_de_pontos_dgps/pontos.txt", sep=";", h=T)
```

```
head(pontos)
```

```
pontos <- pontos[, c("x", "y", "z")]
```

```
coordinates(pontos) <-c("x", "y")
```

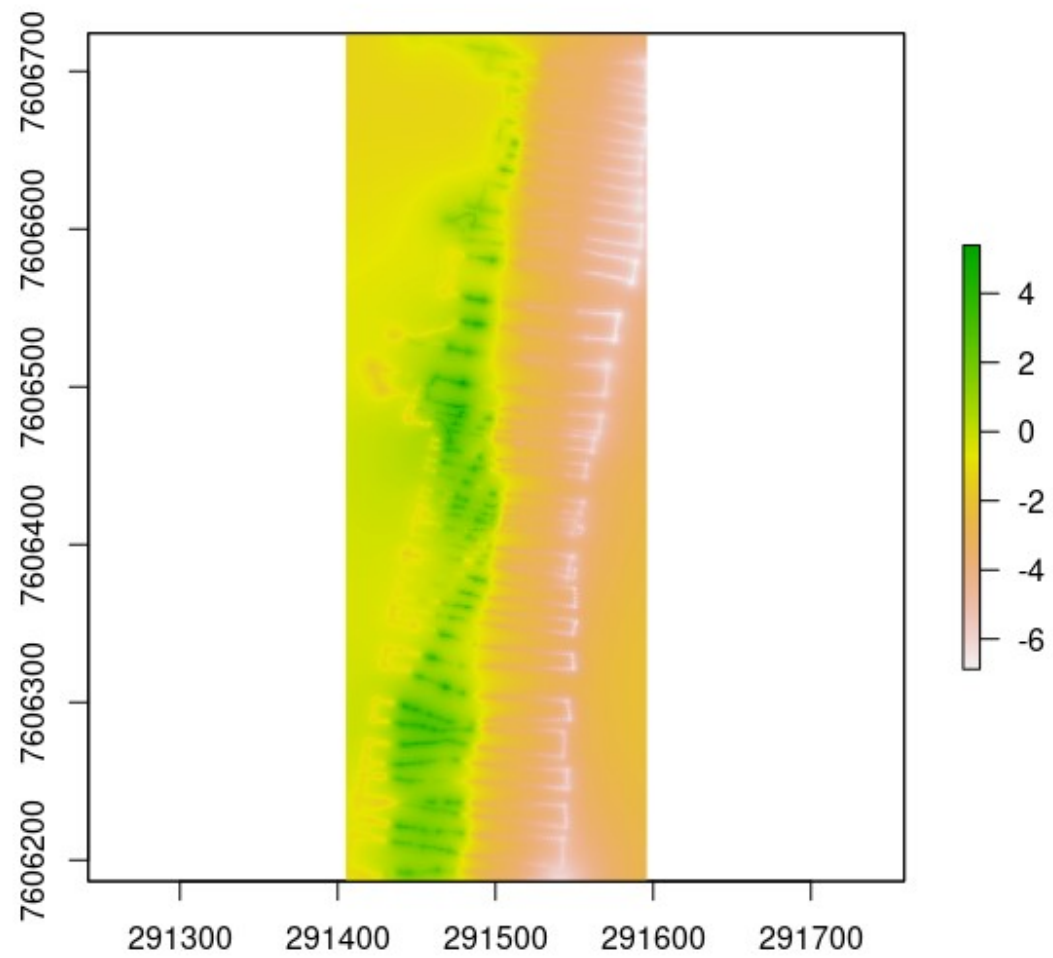
```
#####Interpolação
```

```
r<-raster(extent(pontos), res=0.5) #raster vazio que terá seus valores preenchidos
```

```
idw<-idw(z~1, pontos, as(r, "SpatialGridDataFrame"))
```

```
idw_mde<-raster(as(idw, "SpatialGridDataFrame"))
```

```
writeRaster(idw_mde, "idw_mde.tif", overwrite=T)
```



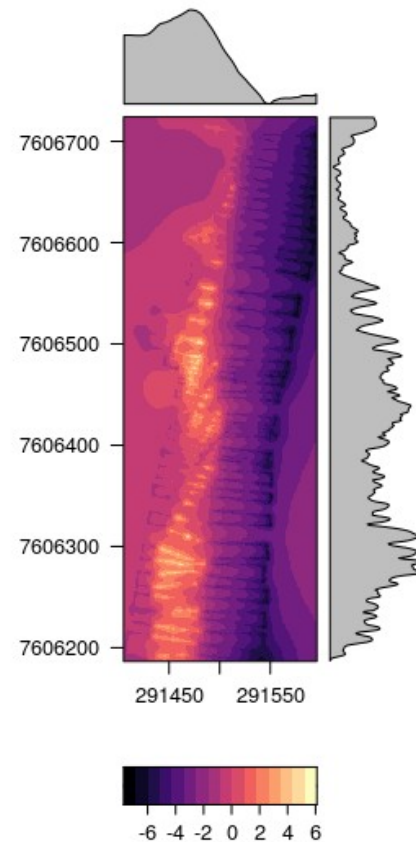


# Plot do MDE

```
install.packages("rasterVis")
```

```
library(rasterVis)
```

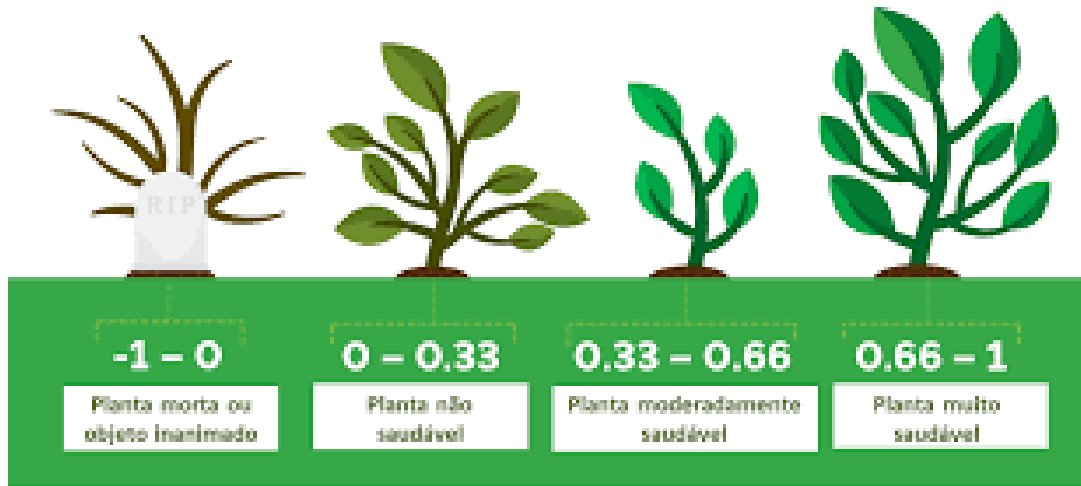
```
levelplot(idw_mde)
```



# Índices de Vegetação

“A vegetation index (VI) is a spectral imaging transformation of two or more image bands designed to enhance the contribution of vegetation properties”

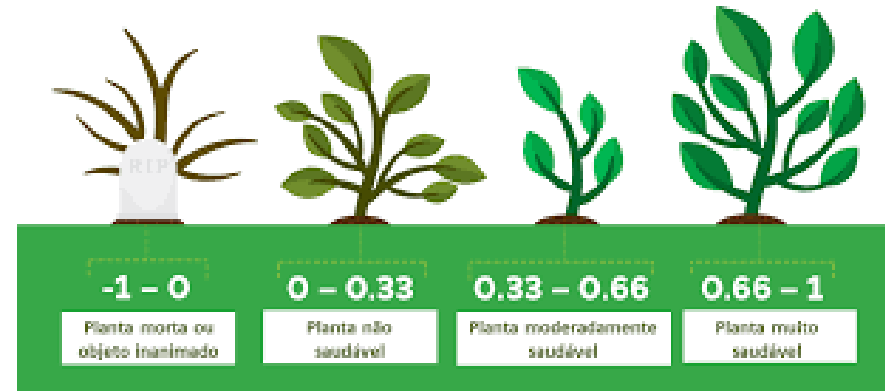
[https://en.wikipedia.org/wiki/Vegetation\\_index](https://en.wikipedia.org/wiki/Vegetation_index)



# NDVI



$$NDVI = \frac{NIR - Red}{NIR + Red}$$



# NDVI

```
library(raster)
```

```
b2<-raster("b2_2011c.tif")
```

```
b3<-raster("b3_2011c.tif")
```

```
b4<-raster("b4_2011c.tif")
```

```
b5<-raster("b5_2011_c.tif")
```

# NDVI

```
niteroi<-shapefile("Bairros_Niteroi.shp")
crs(niteroi)<-CRS("+proj=utm +zone=23 +south +datum=WGS84 +units=m +no_defs +ellps=WGS84 +towgs84=0,0,0") #DEFININDO A PROJEÇÃO
niteroi<-spTransform(niteroi,crs(b3))

b3_m<-mask(b3, niteroi)
plot(b3_m)
b3_m<-crop(b3_m, niteroi)
plot(b3_m)

b4_m<-mask(b4, niteroi)
plot(b4_m)
b4_m<-crop(b4_m, niteroi)
plot(b4_m)

ndvi<-(b4_m-b3_m)/b4_m+b3_m
plot(ndvi)
writeRaster(ndvi, "ndvi.tif")
```

# Exercícios

- 1) Calcule o NDWI
- 2) Implemente funções que façam o cálculo do NDWI e do NDVI.