exercise12

Lee Rui (BA19008042)

2020/5/13

load data

```
library("raster")
```

```
## Warning: package 'raster' was built under R version 3.6.3
```

```
## Loading required package: sp
```

```
## Warning: package 'sp' was built under R version 3.6.3
```

```
library("rgeos")
```

```
## rgeos version: 0.5-2, (SVN revision 621)
##  GEOS runtime version: 3.6.1-CAPI-1.10.1
##  Linking to sp version: 1.4-1
##  Polygon checking: TRUE
```

```
library("rgdal")
```

```
## rgdal: version: 1.4-8, (SVN revision 845)
##  Geospatial Data Abstraction Library extensions to R successfully loaded
##  Loaded GDAL runtime: GDAL 2.2.3, released 2017/11/20
##  Path to GDAL shared files: E:/R-3.6.2/library/rgdal/gdal
##  GDAL binary built with GEOS: TRUE
##  Loaded PROJ.4 runtime: Rel. 4.9.3, 15 August 2016, [PJ_VERSION: 493]
##  Path to PROJ.4 shared files: E:/R-3.6.2/library/rgdal/proj
##  Linking to sp version: 1.4-1
```

```
library("landscapemetrics")
```

```
## Warning: package 'landscapemetrics' was built under R version 3.6.3
```

```
Sys.getenv("GDAL_DATA")
```

```
## [1] "E:/R-3.6.2/library/rgdal/gdal"
```

```
#remotes::install_github("jannes-m/RQGIS3")
library("RQGIS3")
```

```
## Loading required package: reticulate
```

```
## Warning: package 'reticulate' was built under R version 3.6.3
```

```
qgis_session_info()
```

```
## Trying to find QGIS in C:/
```

```
## $gdal
## [1] "3.0.4"
##
## $grass7
## [1] FALSE
##
## $qgis_version
## [1] "3.12.2-Bucure<U+0219>ti"
##
## $saga
## [1] "2.3.2"
```

```
set_env(dev=FALSE)
```

```
## $root
## [1] "C:/Program Files/QGIS 3.12"
##
## $qgis_prefix_path
## [1] "C:/Program Files/QGIS 3.12/apps/qgis"
##
## $python_plugins
## [1] "C:/Program Files/QGIS 3.12/apps/qgis/python/plugins"
##
## $platform
## [1] "Windows"
```

```
#open_app()#Repeat this action will give an error
```

load point data

```
library("tidyverse")
```

```
## Warning: package 'tidyverse' was built under R version 3.6.3
```

```
## -- Attaching packages ----------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.0     v purrr   0.3.3
## v tibble  3.0.1     v dplyr   0.8.5
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0

## Warning: package 'ggplot2' was built under R version 3.6.3

## Warning: package 'tibble' was built under R version 3.6.3

## Warning: package 'tidyr' was built under R version 3.6.3

## Warning: package 'readr' was built under R version 3.6.3

## Warning: package 'purrr' was built under R version 3.6.3

## Warning: package 'dplyr' was built under R version 3.6.3

## Warning: package 'forcats' was built under R version 3.6.3

## -- Conflicts --------------------- tidyverse_conflicts() --
## x tidyr::extract() masks raster::extract()
## x dplyr::filter()  masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## x dplyr::select()  masks raster::select()
```

```
point_table<-read.csv("E:/RStudio/workspace/ecology/GISBEEBOOK_data/QGIS/points_table.csv")
str(point_table)
```

```
## 'data.frame':    20 obs. of  1 variable:
##  $ ID.POINT_X.POINT_Y: Factor w/ 20 levels "1;577573.7938;183596.5398",..: 1 12 14 15 16 17 18 19 20
```

```
point_divided<-point_table %>%separate("ID.POINT_X.POINT_Y", into=c("ID","POINT_X","POINT_Y"),sep=";", 
point.sp<-SpatialPoints(coords = point_divided[,c("POINT_X","POINT_Y")])
point.spdf<-SpatialPointsDataFrame(point_divided[,c("POINT_X","POINT_Y")],data = point_divided[,2:3])
crs(point.sp)
```

```
## CRS arguments: NA
```

```
mycrs<-CRS(projargs="+proj=somerc +lat_0=46.9524055555556 +lon_0=7.43958333333333 +k_0=1 +x_0=600000 +y
proj4string(point.spdf)<-mycrs
point.spdf
```

```
## class       : SpatialPointsDataFrame
## features    : 20
## extent      : 572320.7, 584068.3, 180299.8, 188724.2  (xmin, xmax, ymin, ymax)
## crs         : +proj=somerc +lat_0=46.9524055555556 +lon_0=7.43958333333333 +k_0=1 +x_0=600000 +y_0=20
## variables   : 2
## names       :     POINT_X,     POINT_Y
## min values  : 572320.7458, 180299.8249
## max values  : 584068.2693, 188724.1751
```

```
writeOGR(point.spdf,"E:/RStudio/workspace/ecology/exercise12","point_spdf",driver = "ESRI Shapefile",ov
```

load raster data and reprojection

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

```
g100_06<-raster("E:/RStudio/workspace/ecology/g100_06/g100_06.tif")
find_algorithms(search_term = "Warp")
```

```
## [1] "Warp (reproject)----------------------------------->gdal:warpreproject"
## [2] "Warping shapes------------------------------------->saga:warpingshapes"
```

```
#get_usage(alg = "gdal:warpreproject")
#get_args_man(alg = "gdal:warpreproject", options = TRUE)
#run_qgis(alg = "gdal:warpreproject", INPUT = g100_06, SOURCE_CRS = "EPSG: 21781", TARGET_CRS= "EPSG: 2
g100_06_chCRS<-raster("E:/RStudio/workspace/ecology/exercise12/g100_06_chcrs.tif")
```

load vetor data

```
commune_boundary_FR<-readOGR('E:/RStudio/workspace/ecology/GISBEEBOOK_data/QGIS/commune_boundary_FR.shp
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "E:\RStudio\workspace\ecology\GISBEEBOOK_data\QGIS\commune_boundary_FR.shp", layer: "commune_
## with 51 features
## It has 16 fields
```

```
point<-readOGR("E:/RStudio/workspace/ecology/exercise12/point_spdf.shp")
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "E:\RStudio\workspace\ecology\exercise12\point_spdf.shp", layer: "point_spdf"
## with 20 features
## It has 2 fields
```

clip and mask

```
projection(g100_06_chCRS)
```

```
## [1] "+proj=somerc +lat_0=46.95240555555556 +lon_0=7.439583333333333 +k_0=1 +x_0=600000 +y_0=200000 +e
```

```
projection(point)
```

```
## [1] "+proj=somerc +lat_0=46.9524055555556 +lon_0=7.43958333333333 +k_0=1 +x_0=600000 +y_0=200000 +el
```

```
projection(commune_boundary_FR)
```

```
## [1] "+proj=somerc +lat_0=46.95240555555556 +lon_0=7.439583333333333 +k_0=1 +x_0=600000 +y_0=200000 +
```

```
point_transCRS<-spTransform(point,crs(g100_06_chCRS))
commune_boundary_FR_transCRS<-spTransform(commune_boundary_FR,crs(g100_06_chCRS))

projection(g100_06_chCRS)
```

```
## [1] "+proj=somerc +lat_0=46.95240555555556 +lon_0=7.439583333333333 +k_0=1 +x_0=600000 +y_0=200000 +
```

```
projection(commune_boundary_FR_transCRS)
```

```
## [1] "+proj=somerc +lat_0=46.95240555555556 +lon_0=7.439583333333333 +k_0=1 +x_0=600000 +y_0=200000 +
```

```
projection(point_transCRS)
```

```
## [1] "+proj=somerc +lat_0=46.95240555555556 +lon_0=7.439583333333333 +k_0=1 +x_0=600000 +y_0=200000 +
```

```
extent(point_transCRS)
```

```
## class      : Extent
## xmin       : 572320.7
## xmax       : 584068.3
## ymin       : 180299.8
## ymax       : 188724.2
```

```
extent(g100_06_chCRS)
```

```
## class      : Extent
## xmin       : -2381198
## xmax       : 4090972
## ymin       : -1670049
## ymax       : 3190337
```

```
extent(commune_boundary_FR_transCRS)
```

```
## class      : Extent
## xmin       : 564835
## xmax       : 593200
## ymin       : 168930
## ymax       : 196915
```

```
CLC06_Fribourg<-crop(g100_06_chCRS,commune_boundary_FR_transCRS)
CLC06_Fribourg_maskedch<-mask(CLC06_Fribourg,commune_boundary_FR_transCRS,updateNA=TRUE)
writeRaster(CLC06_Fribourg_maskedch,"E:/RStudio/workspace/ecology/exercise12/CLC06_Fribourg_masked.tif"
```

add buffer(2000m) and visaulization

```
#raster::extract(CLC06_Fribourg_maskedch,point_transCRS,buffer=2000)
find_algorithms(search_term = "Buffer", name_only = TRUE)
```

```
## [1] "gdal:buffervectors" "native:buffer"
```

```
#get_usage(alg = "native:buffer")
#get_args_man(alg = "native:buffer", options = TRUE)
run_qgis(alg = "native:buffer", INPUT = point_transCRS, DISTANCE = 2000, OUTPUT= "E:/RStudio/workspace/
```

```
## $OUTPUT
## [1] "E:/RStudio/workspace/ecology/exercise12/buffered.shp"
```

```
buffer<-readOGR("E:/RStudio/workspace/ecology/exercise12/buffered.shp")
```

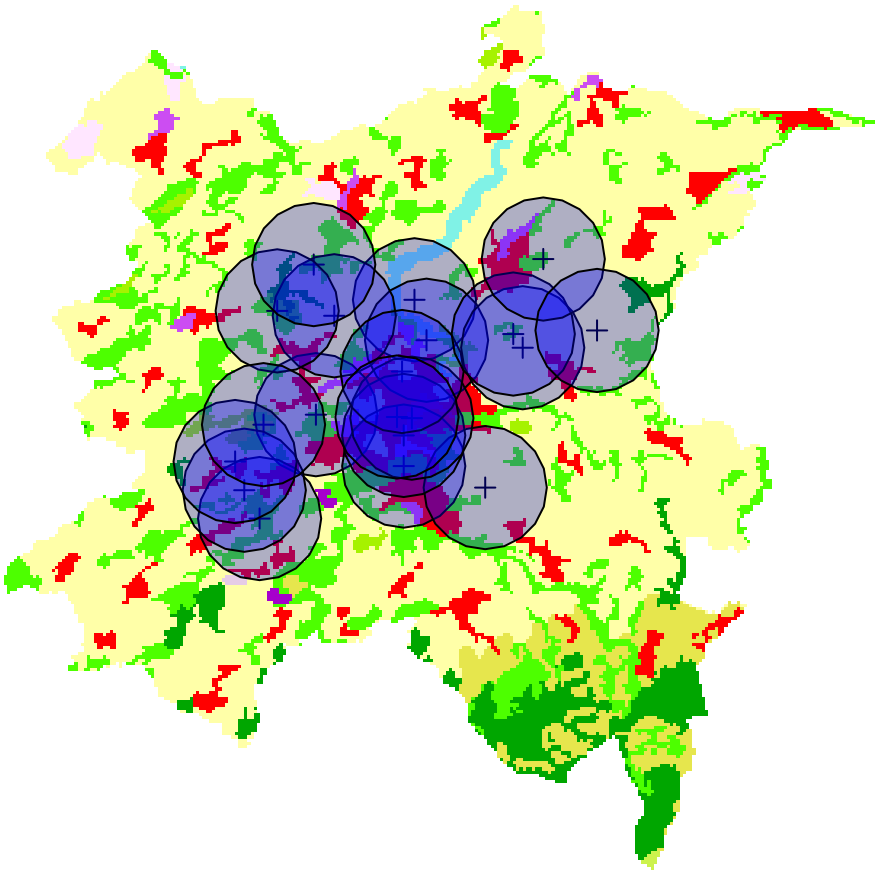
```
## OGR data source with driver: ESRI Shapefile
## Source: "E:\RStudio\workspace\ecology\exercise12\buffered.shp", layer: "buffered"
## with 20 features
## It has 2 fields
```

```
plot(CLC06_Fribourg_maskedch)
```

```r
plot(point_transCRS,type = "o", col = "black",add=TRUE)
plot(buffer,type = "o",col = rgb(0, 0, 255, 80, maxColorValue=255),add=TRUE)
```

```
## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete

## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete

## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete

## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete

## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete

## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete

## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete

## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete

## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete

## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete

## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete

## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete

## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete

## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete

## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete

## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete

## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete
```

```
## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete

## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete

## Warning in polypath(x = mcrds[, 1], y = mcrds[, 2], border = border, col =
## col, : graphical parameter "type" is obsolete
```



calculating proportion and add label

```
POint_buffer<-as(buffer,"SpatialPolygons")
startc=c()
for (i in 1:20){
  POint_bufferm<-mask(CLC06_Fribourg,POint_buffer[i],updateNA=TRUE)
  POint_buffermv<-values(POint_bufferm)
  class12<-sum(POint_buffermv==12,na.rm=TRUE)/(length(POint_buffermv)-sum(is.na(POint_buffermv)))
  class24<-sum(POint_buffermv==24,na.rm=TRUE)/(length(POint_buffermv)-sum(is.na(POint_buffermv)))
  class25<-sum(POint_buffermv==25,na.rm=TRUE)/(length(POint_buffermv)-sum(is.na(POint_buffermv)))
  classother<-1-class12-class24-class25
  startc<-append(startc,c(class12,class24,class25,classother))
```

```
}
Land_proportion<-data.frame(t(matrix(c(startc),nrow=4)))
```

```
legend<-read.delim("E:/RStudio/workspace/ecology/g100_06/clc_legend_qgis.txt",skip=1)
str(legend)
```

```
## 'data.frame':    48 obs. of  1 variable:
##  $ INTERPOLATION.DISCRETE: Factor w/ 48 levels "1,230,000,077,255, 111 - Continuous urban fabric",..
```

```
clc_legend<-separate(legend,"INTERPOLATION.DISCRETE",into = c("GRID_CODE","b","c","d","e","f"),sep = ",
```

```
## Warning: Expected 2 pieces. Additional pieces discarded in 1 rows [30].
```

```
clc_legend
```

```
##    GRID_CODE              RGB CLC_CODE
## 1          1  230,000,077,255      111
## 2          2  255,000,000,255      112
## 3          3  204,077,242,255      121
## 4          4  204,000,000,255      122
## 5          5  230,204,204,255      123
## 6          6  230,204,230,255      124
## 7          7  166,000,204,255      131
## 8          8  166,077,000,255      132
## 9          9 255,077,255, 255      133
## 10        10  255,166,255,255      141
## 11        11  255,230,255,255      142
## 12        12  255,255,168,255      211
## 13        13  255,255,000,255      212
## 14        14  230,230,000,255      213
## 15        15  230,128,000,255      221
## 16        16  242,166,077,255      222
## 17        17  230,166,000,255      223
## 18        18  230,230,077,255      231
## 19        19  255,230,166,255      241
## 20        20  255,230,077,255      242
## 21        21  230,204,077,255      243
## 22        22  242,204,166,255      244
## 23        23  128,255,000,255      311
## 24        24  000,166,000,255      312
## 25        25  077,255,000,255      313
## 26        26  204,242,077,255      321
## 27        27  166,255,128,255      322
## 28        28  166,230,077,255      323
## 29        29  166,242,000,255      324
## 30        30  230,230,230,255      331
## 31        31  204,204,204,255      332
## 32        32  204,255,204,255      333
## 33        33  000,000,000,255      334
## 34        34  166,230,204,255      335
## 35        35  166,166,255,255      411
```

```
## 36        36  077,077,255,255      412
## 37        37  204,204,255,255      421
## 38        38  230,230,255,255      422
## 39        39  166,166,230,255      423
## 40        40  000,204,242,255      511
## 41        41  128,242,230,255      512
## 42        42  000,255,166,255      521
## 43        43  166,255,230,255      522
## 44        44  230,242,255,255      523
## 45        48  255,255,255,255      999
## 46        49  255,255,255,255      990
## 47        50  230,242,255,255      995
## 48       255  255,255,255,255      990
##                                                                                LABEL
## 1                                                                 Continuous urban fabric
## 2                                                               Discontinuous urban fabric
## 3                                                           Industrial or commercial units
## 4                                                 Road and rail networks and associated land
## 5                                                                        Port areas
## 6                                                                          Airports
## 7                                                             Mineral extraction sites
## 8                                                                        Dump sites
## 9                                                                 Construction sites
## 10                                                                 Green urban areas
## 11                                                       Sport and leisure facilities
## 12                                                          Non-irrigated arable land
## 13                                                        Permanently irrigated land
## 14                                                                       Rice fields
## 15                                                                         Vineyards
## 16                                                  Fruit trees and berry plantations
## 17                                                                       Olive groves
## 18                                                                          Pastures
## 19                                               Annual crops associated with permanent crops
## 20                                                         Complex cultivation patterns
## 21 Land principally occupied by agriculture with significant areas of natural vegetation
## 22                                                                 Agro-forestry areas
## 23                                                                 Broad-leaved forest
## 24                                                                   Coniferous forest
## 25                                                                        Mixed forest
## 26                                                                 Natural grasslands
## 27                                                                 Moors and heathland
## 28                                                               Sclerophyllous vegetation
## 29                                                             Transitional woodland-shrub
## 30                                                                         Beaches
## 31                                                                       Bare rocks
## 32                                                            Sparsely vegetated areas
## 33                                                                       Burnt areas
## 34                                                       Glaciers and perpetual snow
## 35                                                                  Inland marshes
## 36                                                                        Peat bogs
## 37                                                                     Salt marshes
## 38                                                                           Salines
## 39                                                                 Intertidal flats
## 40                                                                   Water courses
```

```
## 41                                                                  Water bodies
## 42                                                                Coastal lagoons
## 43                                                                     Estuaries
## 44                                                                  Sea and ocean
## 45                                                                        NODATA
## 46                                                       UNCLASSIFIED LAND SURFACE
## 47                                                       UNCLASSIFIED WATER BODIES
## 48                                                                  UNCLASSIFIED
```

```
names(Land_proportion)<-c(gsub("-", "_", gsub(" ", "_", clc_legend[which(clc_legend == 12, arr.ind=TRUE
```

```
Landcoor_proportion<-cbind(point_divided,Land_proportion)
Landcoor_proportion#The coordinate deviation of the read-in and read-out data in R leads to a slight de
```

```
##    ID  POINT_X  POINT_Y Non_irrigated_arable_land Coniferous_forest
## 1   1 577573.8 183596.5                 0.1817391       0.000000000
## 2   2 574942.2 183671.9                 0.4882302       0.000000000
## 3   3 577793.1 182003.7                 0.1914336       0.000000000
## 4   4 580438.2 181305.2                 0.7200350       0.000000000
## 5   5 578141.6 187401.3                 0.6846690       0.000000000
## 6   6 578533.2 186088.9                 0.5619546       0.000000000
## 7   7 575538.1 186882.7                 0.6652098       0.049825175
## 8   8 573686.0 187041.4                 0.6687063       0.082167832
## 9   9 574871.3 188544.3                 0.7027027       0.081952921
## 10 10 581655.3 185845.5                 0.8713911       0.000000000
## 11 11 573114.5 180299.8                 0.5875657       0.000000000
## 12 12 572617.1 181220.6                 0.5986038       0.001745201
## 13 13 572320.7 182151.9                 0.5698080       0.018324607
## 14 14 573241.5 183347.8                 0.5817223       0.000000000
## 15 15 577792.3 182998.6                 0.1576655       0.000000000
## 16 16 578056.9 183559.5                 0.1881533       0.000000000
## 17 17 577739.4 185072.9                 0.3193351       0.000000000
## 18 18 581348.3 186290.0                 0.8423345       0.000000000
## 19 19 582322.0 188724.2                 0.7332171       0.000000000
## 20 20 584068.3 186406.4                 0.8067885       0.044386423
##    Mixed_forest     Others
## 1    0.12956522 0.68869565
## 2    0.15780296 0.35396687
## 3    0.27185315 0.53671329
## 4    0.13998250 0.13998250
## 5    0.08710801 0.22822300
## 6    0.05584642 0.38219895
## 7    0.18531469 0.09965035
## 8    0.16783217 0.08129371
## 9    0.20052310 0.01482127
## 10   0.04461942 0.08398950
## 11   0.21978984 0.19264448
## 12   0.20855148 0.19109948
## 13   0.17015707 0.24171030
## 14   0.18189807 0.23637961
## 15   0.19163763 0.65069686
## 16   0.14808362 0.66376307
## 17   0.08748906 0.59317585
```

```
## 18   0.02787456 0.12979094
## 19   0.07585004 0.19093287
## 20   0.09399478 0.05483029
```

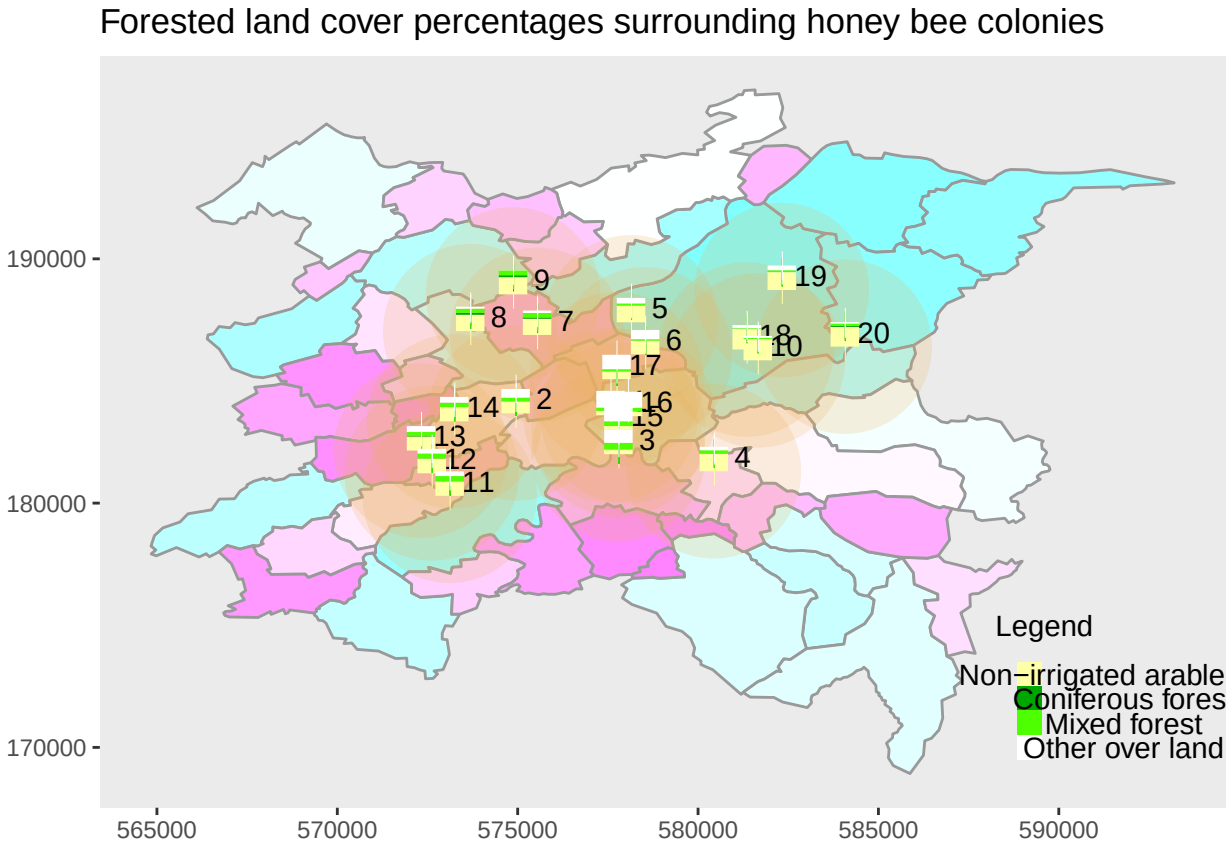
visaulization the Land_proportion of points

```
boundary_df<-fortify(commune_boundary_FR_transCRS)
```

```
## Regions defined for each Polygons
```

```
prov_coor<-cbind(boundary_df[which(boundary_df == 15, arr.ind=TRUE)[1:51,1],1:2],commune_boundary_FR_tra
#prov_coor#Messy coding......
```

```
le<-data.frame(c(589200,589200,589200,589200),c(173000,172000,171000,170000),c("Non-irrigated arable lan
library("ggplot2")
ggplot()+geom_polygon(aes(x=long, y=lat, group=group), data=boundary_df, fill=cm.colors(2640:2640), colo
  #geom_text(aes(x=long+1000, y=lat,label=paste(commune_boundary_FR_transCRS$GEMNAME)),col=terrain.colo
  theme(panel.grid = element_blank(), axis.title = element_blank())+
  geom_point(aes(x=POINT_X, y=POINT_Y),alpha=0.2,colour=terrain.colors(10)[7],size = 30,data=point_divi
  geom_text(aes(x=POINT_X+800, y=POINT_Y+600, label=paste(ID)), data=point_divided)+
  geom_errorbar(aes(x=POINT_X, ymin=POINT_Y, ymax=POINT_Y+Non_irrigated_arable_land*1000), data=Landcoo
  geom_errorbar(aes(x=POINT_X, ymin=POINT_Y+Non_irrigated_arable_land*1000, ymax=POINT_Y+Non_irrigated_a
  geom_errorbar(aes(x=POINT_X, ymin=POINT_Y+Non_irrigated_arable_land*1000+Coniferous_forest*1000, ymax=
  geom_errorbar(aes(x=POINT_X, ymin=POINT_Y+Non_irrigated_arable_land*1000+Coniferous_forest*1000+Mixed_
  ggtitle('Forested land cover percentages surrounding honey bee colonies')+
  geom_text(aes(x=589600, y=175000,label=paste("Legend")),size=4)+
  geom_point(aes(x=le[,1],y=le[,2]),shape=15,colour=c("#ffffa8","#00a600","#4dff00","#ffffff"),size = 4
  geom_text(aes(x=le[,1]+2600, y=le[,2], label=paste(le[,3])), data= le)#(really can't draw the pie)
```



The following is a model of random guessing

preProcess data for model

```
Land_proportion$Major_land_types<-apply(Land_proportion, 1, function(x){which.max(x)})
Land_proportion<-Land_proportion%>%mutate(Major_land_types = case_when(Major_land_types == 1 ~ 'Non_irri
Land_coor_types<-cbind(point_divided[,2:3],Land_proportion[5])
Land_coor_types
```

```
##     POINT_X  POINT_Y          Major_land_types
## 1  577573.8 183596.5                    Others
## 2  574942.2 183671.9 Non_irrigated_arable_land
## 3  577793.1 182003.7                    Others
## 4  580438.2 181305.2 Non_irrigated_arable_land
## 5  578141.6 187401.3 Non_irrigated_arable_land
## 6  578533.2 186088.9 Non_irrigated_arable_land
## 7  575538.1 186882.7 Non_irrigated_arable_land
## 8  573686.0 187041.4 Non_irrigated_arable_land
## 9  574871.3 188544.3 Non_irrigated_arable_land
## 10 581655.3 185845.5 Non_irrigated_arable_land
## 11 573114.5 180299.8 Non_irrigated_arable_land
## 12 572617.1 181220.6 Non_irrigated_arable_land
```

```
## 13 572320.7 182151.9 Non_irrigated_arable_land
## 14 573241.5 183347.8 Non_irrigated_arable_land
## 15 577792.3 182998.6                   Others
## 16 578056.9 183559.5                   Others
## 17 577739.4 185072.9                   Others
## 18 581348.3 186290.0 Non_irrigated_arable_land
## 19 582322.0 188724.2 Non_irrigated_arable_land
## 20 584068.3 186406.4 Non_irrigated_arable_land
```

build rf model

```
library("caret")
```

```
## Warning: package 'caret' was built under R version 3.6.3
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

```
set.seed(521)
train_idx <- createDataPartition(Land_coor_types$Major_land_types, p=0.75, list=FALSE)
training <- Land_coor_types[train_idx,]
test <- Land_coor_types[-train_idx,]

set.seed(521)
uneLength_ctrl <- trainControl(
  method = 'cv',
  number = 10,
  savePredictions = 'final',
  classProbs = T,
  summaryFunction=twoClassSummary)

rf_fit <- train(as.factor(Major_land_types) ~.,
                data = training,
                method = "rf",
                trControl = uneLength_ctrl,
                verbose = FALSE)
```

```
## note: only 1 unique complexity parameters in default grid. Truncating the grid to 1 .
```

```
## Warning in train.default(x, y, weights = w, ...): The metric "Accuracy" was not
## in the result set. ROC will be used instead.
```

```
## Warning in nominalTrainWorkflow(x = x, y = y, wts = weights, info = trainInfo, :
## There were missing values in resampled performance measures.
```

evaluate rf performance

```
rf_pred <- predict(rf_fit, test)
rf_pred
```

```
## [1] Non_irrigated_arable_land Non_irrigated_arable_land
## [3] Non_irrigated_arable_land Others
## Levels: Non_irrigated_arable_land Others
```

```
confusionMatrix(reference = as.factor(test$Major_land_types),
                data = rf_pred,
                mode = "everything")
```

```
## Confusion Matrix and Statistics
##
##                            Reference
## Prediction                  Non_irrigated_arable_land Others
##   Non_irrigated_arable_land                         3      0
##   Others                                            0      1
##
##                Accuracy : 1
##                  95% CI : (0.3976, 1)
##     No Information Rate : 0.75
##     P-Value [Acc > NIR] : 0.3164
##
##                   Kappa : 1
##
##  Mcnemar's Test P-Value : NA
##
##             Sensitivity : 1.00
##             Specificity : 1.00
##          Pos Pred Value : 1.00
##          Neg Pred Value : 1.00
##               Precision : 1.00
##                  Recall : 1.00
##                      F1 : 1.00
##              Prevalence : 0.75
##          Detection Rate : 0.75
##    Detection Prevalence : 0.75
##       Balanced Accuracy : 1.00
##
##        'Positive' Class : Non_irrigated_arable_land
##
```

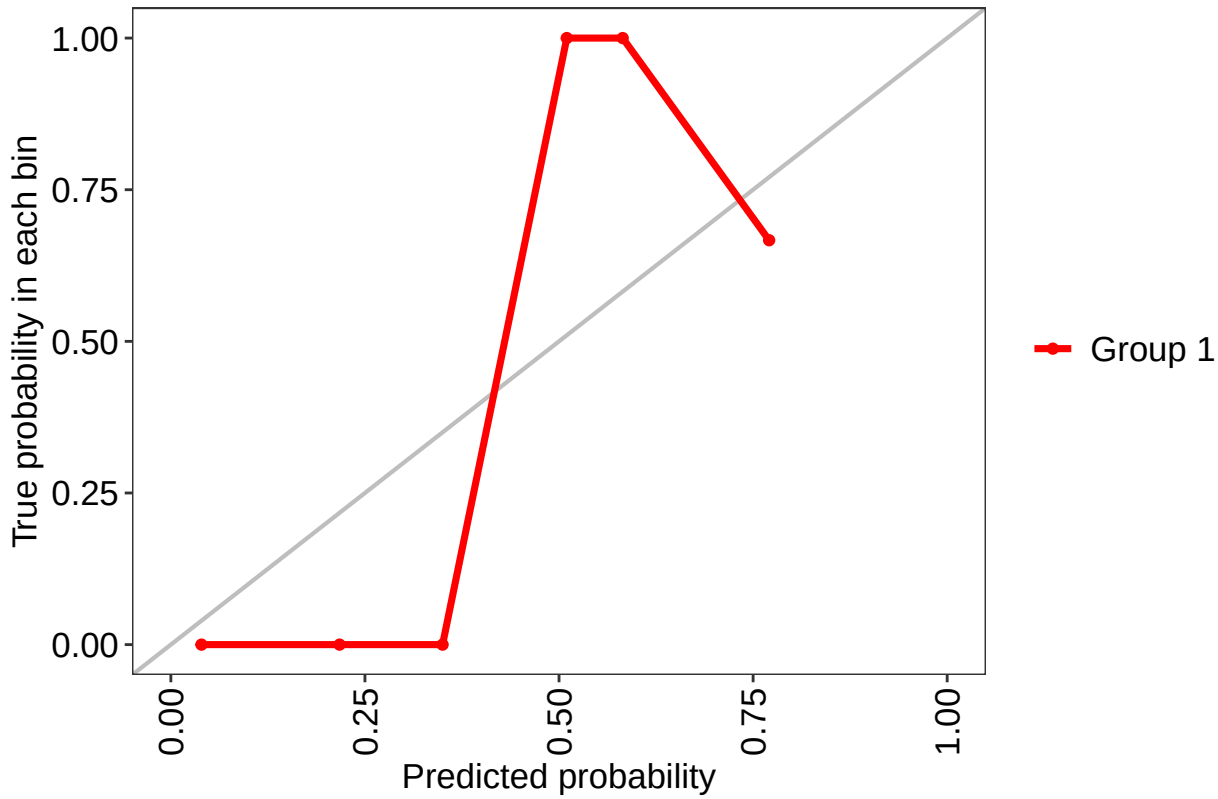
```
library(MLeval)
```

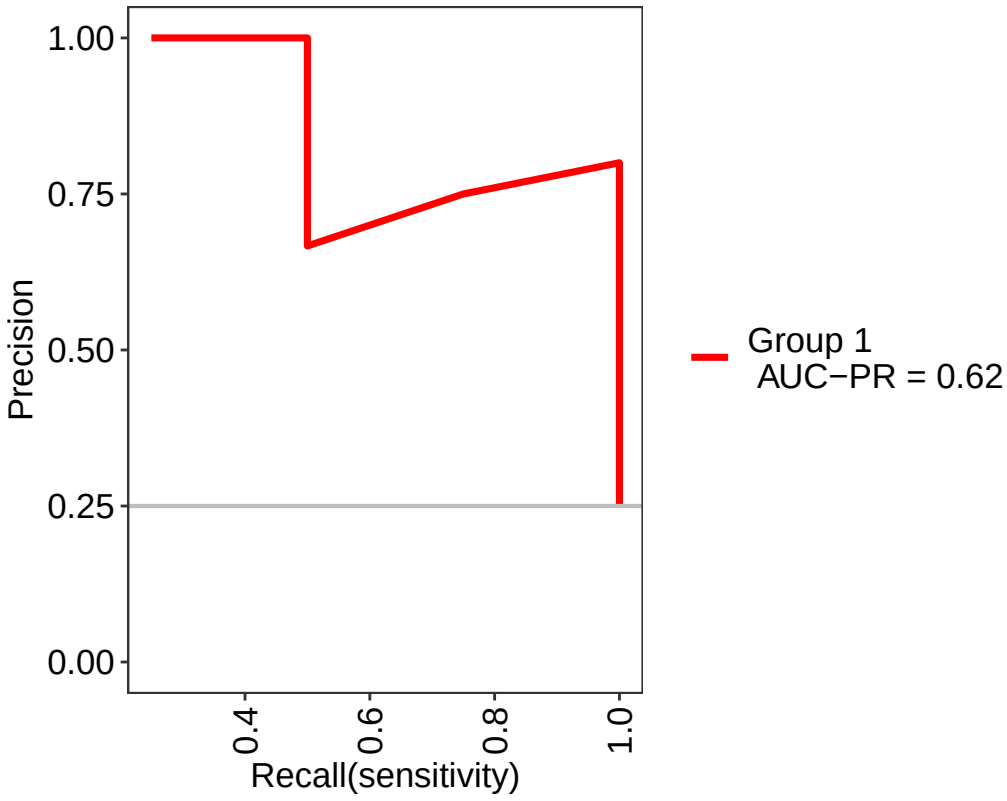
```
## Warning: package 'MLeval' was built under R version 3.6.3
```

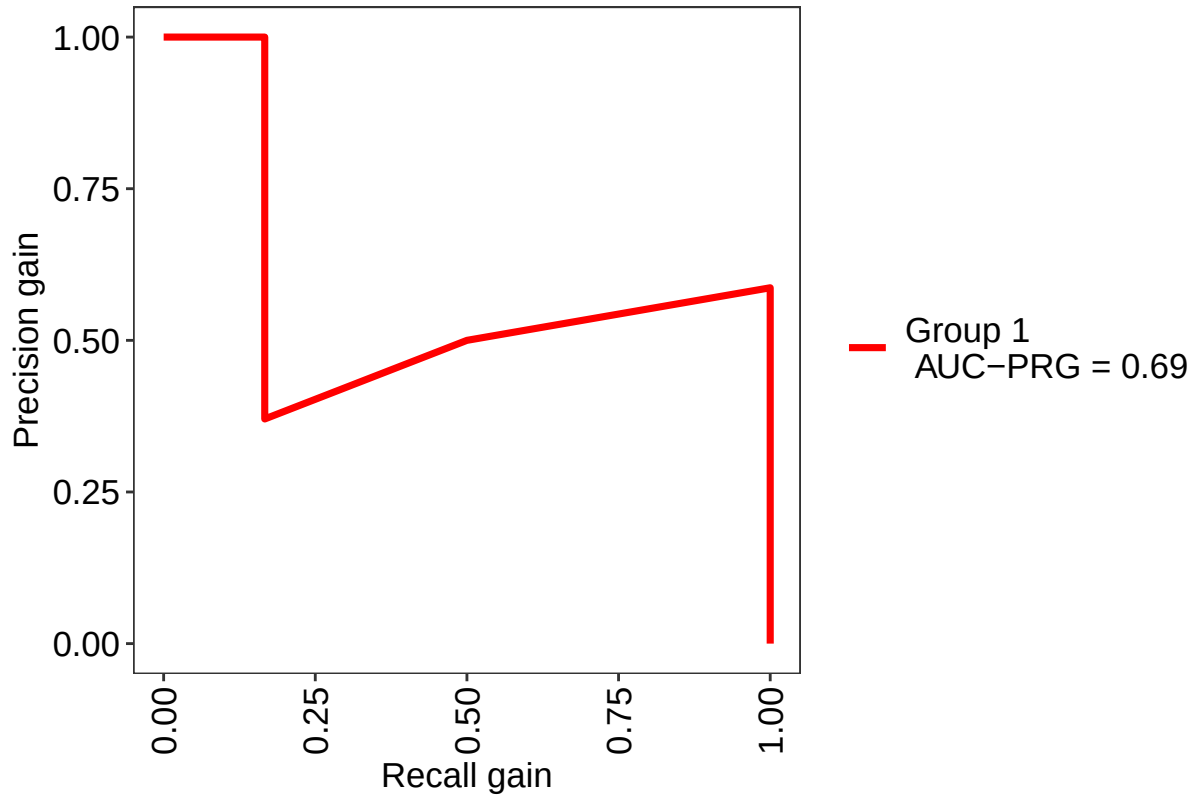
```
x <- evalm(rf_fit)
```

```
## ***MLeval: Machine Learning Model Evaluation***
```

```
## Input: caret train function object

## Not averaging probs.

## Group 1 type: cv

## Observations: 16

## Number of groups: 1

## Observations per group: 16

## Positive: Others

## Negative: Non_irrigated_arable_land

## Group: Group 1

## Positive: 4

## Negative: 12

## ***Performance Metrics***
```

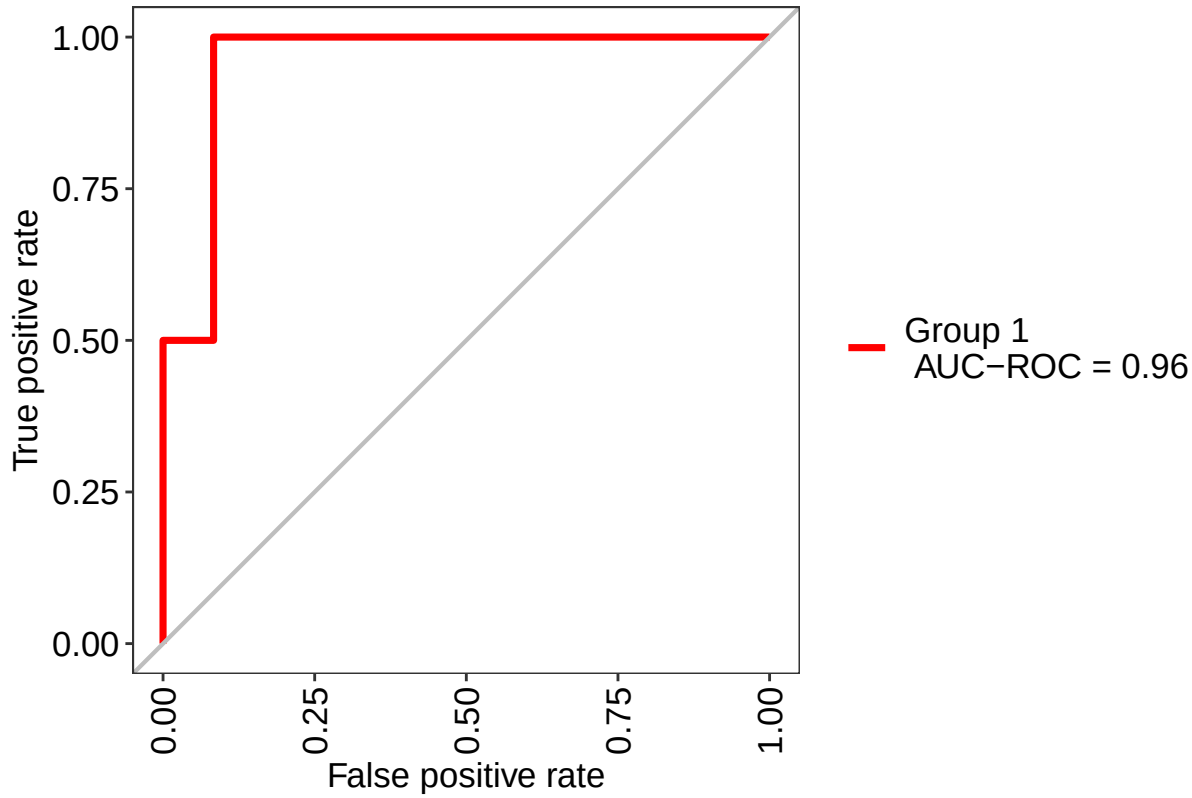



1.00
0.75
0.50
0.25
0.00
Precision
0.4
0.6
0.8
1.0
Recall(sensitivity)
Group 1
AUC−PR = 0.62

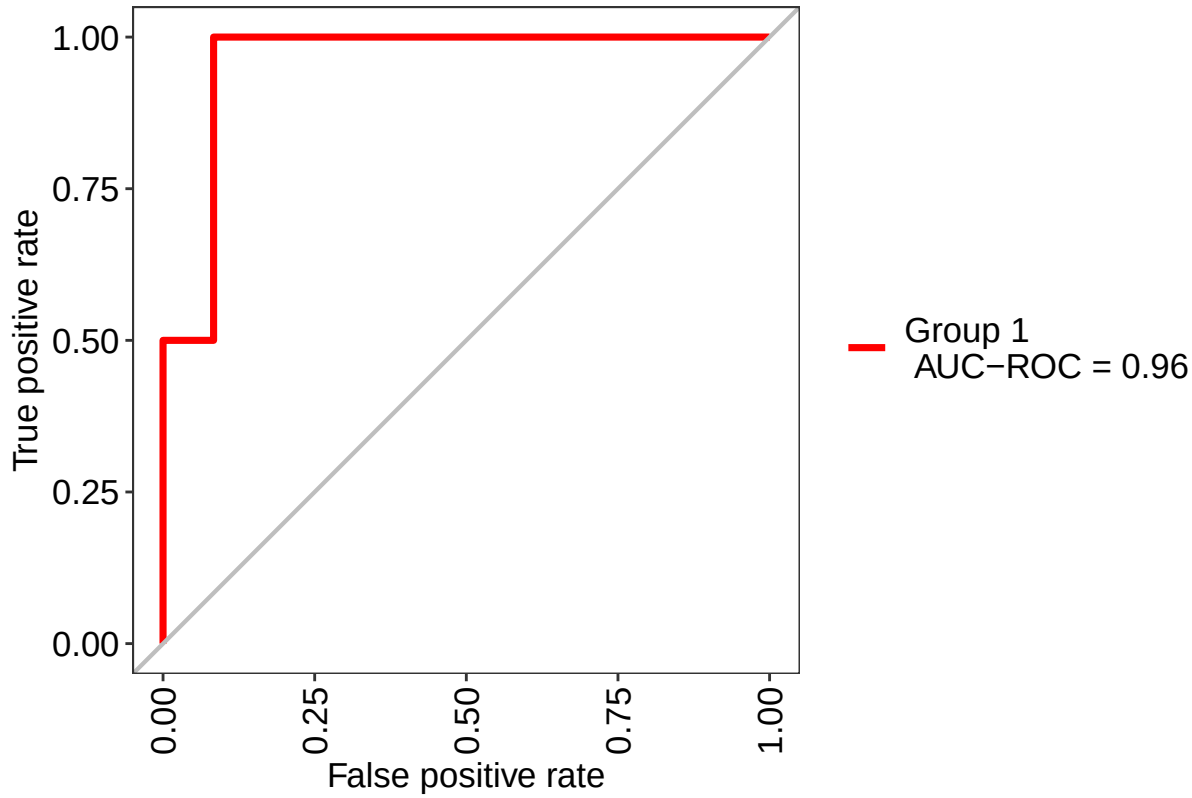



```
## Group 1 Optimal Informedness = 0.916666666666667
```

```
## Group 1 AUC-ROC = 0.96
```


1.00
0.75
0.50
0.25
0.00
True positive rate
0.00
0.25
0.50
0.75
1.00
False positive rate
Group 1
AUC−ROC = 0.96

```
x$roc
```




```
x$stdres
```

```
## $`Group 1`
##                Score        CI
## SENS           1.000    0.51-1
## SPEC           0.917 0.65-0.99
## MCC            0.856      <NA>
## Informedness   0.917      <NA>
## PREC           0.800 0.38-0.96
## NPV            1.000    0.74-1
## FPR            0.083      <NA>
## F1             0.889      <NA>
## TP             4.000      <NA>
## FP             1.000      <NA>
## TN            11.000      <NA>
## FN             0.000      <NA>
## AUC-ROC        0.960  0.82-1.1
## AUC-PR         0.620      <NA>
## AUC-PRG        0.690      <NA>
```