

Geozoning structures

B. Charnomordic

2017-09-14

Contents

Structure 1: map	1
Structure 2: zoning geometry	3
Structure 3: zoning structure	5

```
library(geozoning)
library(sp)
library(fields)
library(gstat)
```

This vignette describes the main structures used by the geozoning package: map, zoning object and zoning geometry, how to create and use them.

Structure 1: map

A map structure contains the data and parameters necessary to perform a zoning. The structure is identical whether the data are simulated or not. The map structure contains raw data for traceability: rawData component (class SpatialPointsDataFrame) and kriged data in 2 useful forms: krigData component (class SpatialPointsDataFrame) used for zoning calculations and krigGrid component (class matrix), used for image plot displays. The resolution grid is given in the step component. Kriged data are normalized so that x-coordinates are between 0 and 1. y-coordinates are normalized with the same ratio used for x-coordinates. The ratio is recorded in the ratio component. Kriging is either done with inverse distance interpolation, or with a variogram model fitted to data, which is recorded in 2 components: VGMmodel (variogramModel) and modelGen (class RMmodel). The boundary component is a list with x and y components, normalized with the same ratio used for data, corresponding to the map boundary. The map structure also contains the list of neighbours of each kriged data point: krigN. For each kriged data point, the corresponding list element contains the sorted indexes of all its neighbors (sharing an edge in the kriged point Voronoi tessellation). The list of areas of kriged Voronoi polygons is given in the krigSurfVoronoi component. It is used to assign weights in calculations.

The map object can be created by 2 functions: genMap or randKmap.

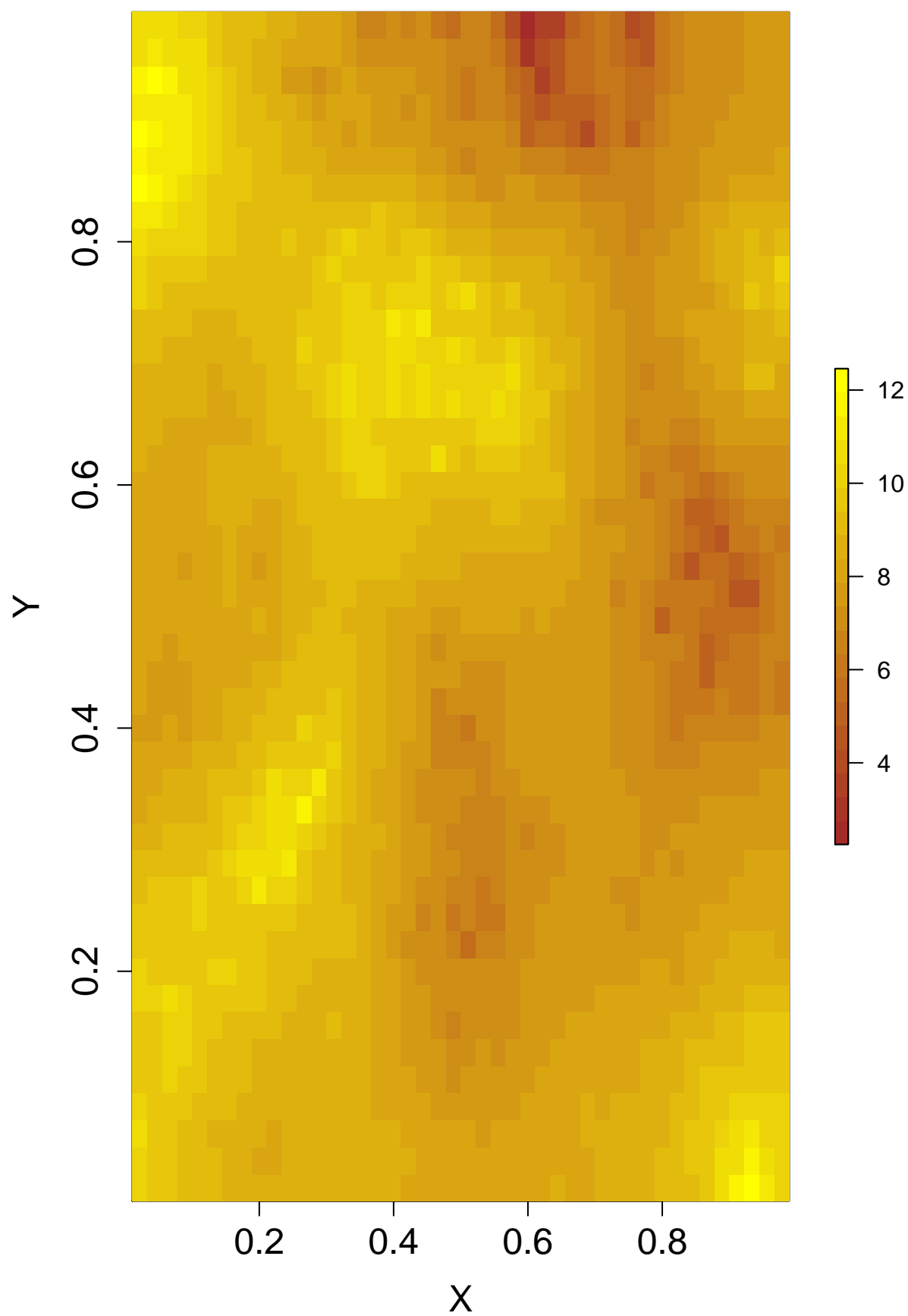
```
seed=80
map=genMap(DataObj=NULL,seed=seed,disp=FALSE,Vmean=15,krig=2)

## [1] "DataObj=NULL, generating DataObj-seed= 80"
## [inverse distance weighted interpolation]

# or
map = randKmap(DataObj=NULL,nPointsK=500,Vmean=10,krig=1)

## [1] "DataObj=NULL, generating DataObj-seed= "
## [using ordinary kriging]

Display map with kriged data
dispZ(mapTest$step,matVal=mapTest$krigGrid)
```



```
## NULL
```

Check the mean and standard deviation of generated data.

```
meanvarSimu(map)
```

```
##      raw mean kriged mean      raw sd   kriged sd
##  11.172385   11.076660   2.383384   2.370930
```

Structure 2: zoning geometry

A zoning geometry *Z* contains the polygons corresponding to the boundaries of zones within a zoning. For size limiting considerations, it contains no other data (see zoning structure that contains data and zone geometry). *Z* is a list of *SpatialPolygons*. Some zones may have holes, in that case there is always an additional *SpatialPolygons* corresponding to the hole.

Z can be created or updated by many functions: *initialZoning*, *correctionTree*, *zoneFusion2*, *optiGrow*, etc.

```
data(mapTest)
ZK=initialZoning(qProb=c(0.4,0.7),mapTest)
Z=ZK$resZ$zonePolygone
class(Z)
```

```
## [1] "list"
```

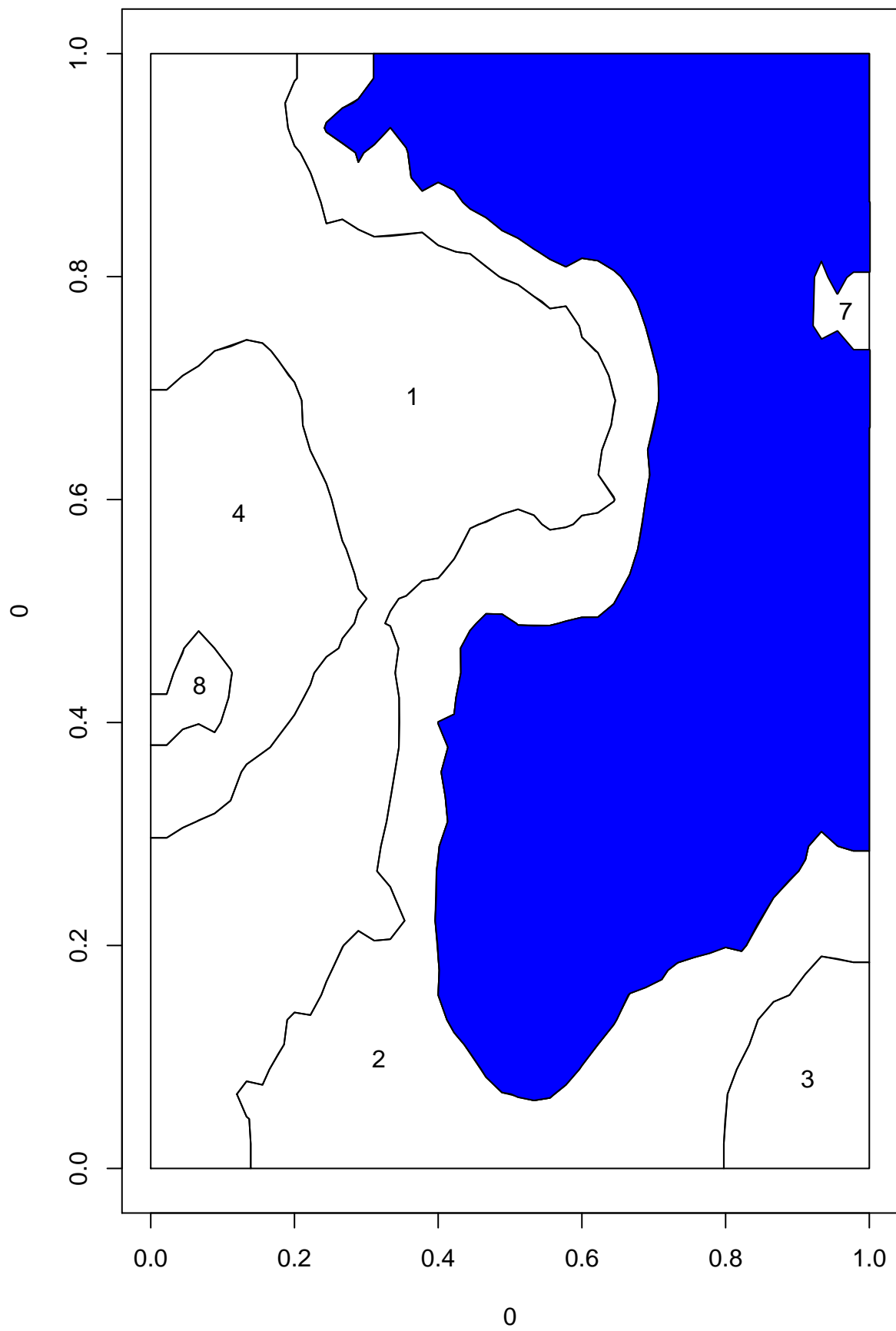
```
plotZ(Z)
```

```
## NULL
```

```
Zf=zoneFusion2(Z[[5]],Z[[6]])
class(Zf)
```

```
## [1] "SpatialPolygons"
## attr(,"package")
## [1] "sp"
```

```
sp::plot(Zf,add=TRUE,col="blue")
```



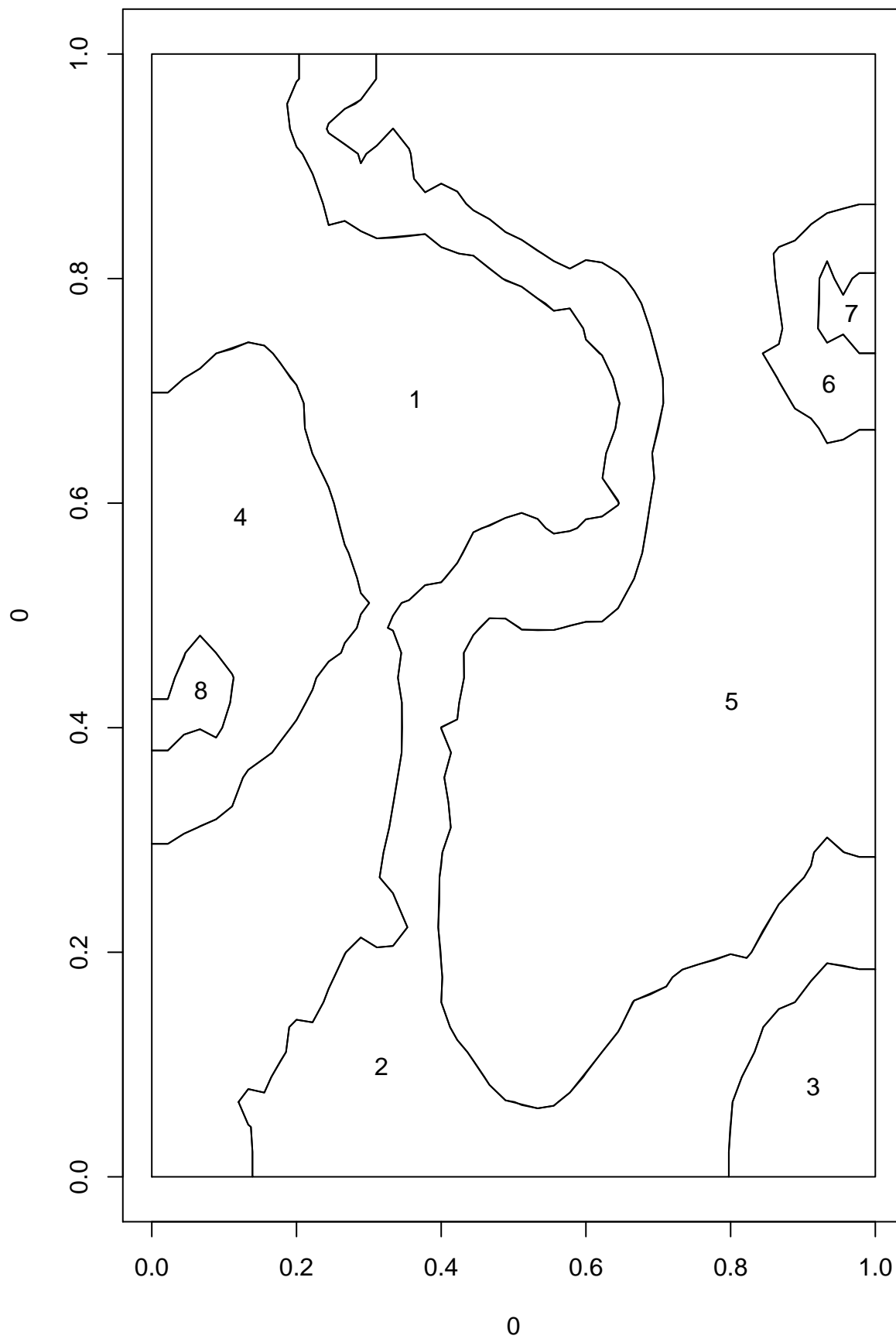
Structure 3: zoning structure

A zoning structure K contains all elements resulting from a zoning: zoning geometry (zonePolygone component - list of SpatialPolygons), zone neighborhood (zoneN or zoneNmodif with no self-neighborhood), data point zone assignment, zone areas and mean values. It is created by the calNei function, that is called by many functions, for instance from within initialZoning or correctedTree.

```
data(mapTest)
ZK=initialZoning(qProb=c(0.4,0.7),mapTest)
K=ZK$resZ
names(K)

## [1] "zoneN"          "zoneNModif"      "listZonePoint"  "meanTot"
## [5] "critSurf"       "meanZone"        "listSurf"       "zonePolygone"
## [9] "lab"            "qProb"

Z=K$zonePolygone
plotZ(Z)
```



```
## NULL
```

The labZone function may be called to assign zone labels depending on the zone mean value as follows. Default label is 1, corresponding to a mean value smaller or equal to first quantile. For p ordered quantile values, if mean value is greater than quantile k and smaller or equal to quantile k+1, zone label is k+1. if mean value is greater than quantile p, zone label is p+1.

```
p = K$qProb  
K=labZone(K,p,mapTest$krigGrid)  
print(K$lab)
```

```
## [1] 3 2 3 2 1 2 3 1
```

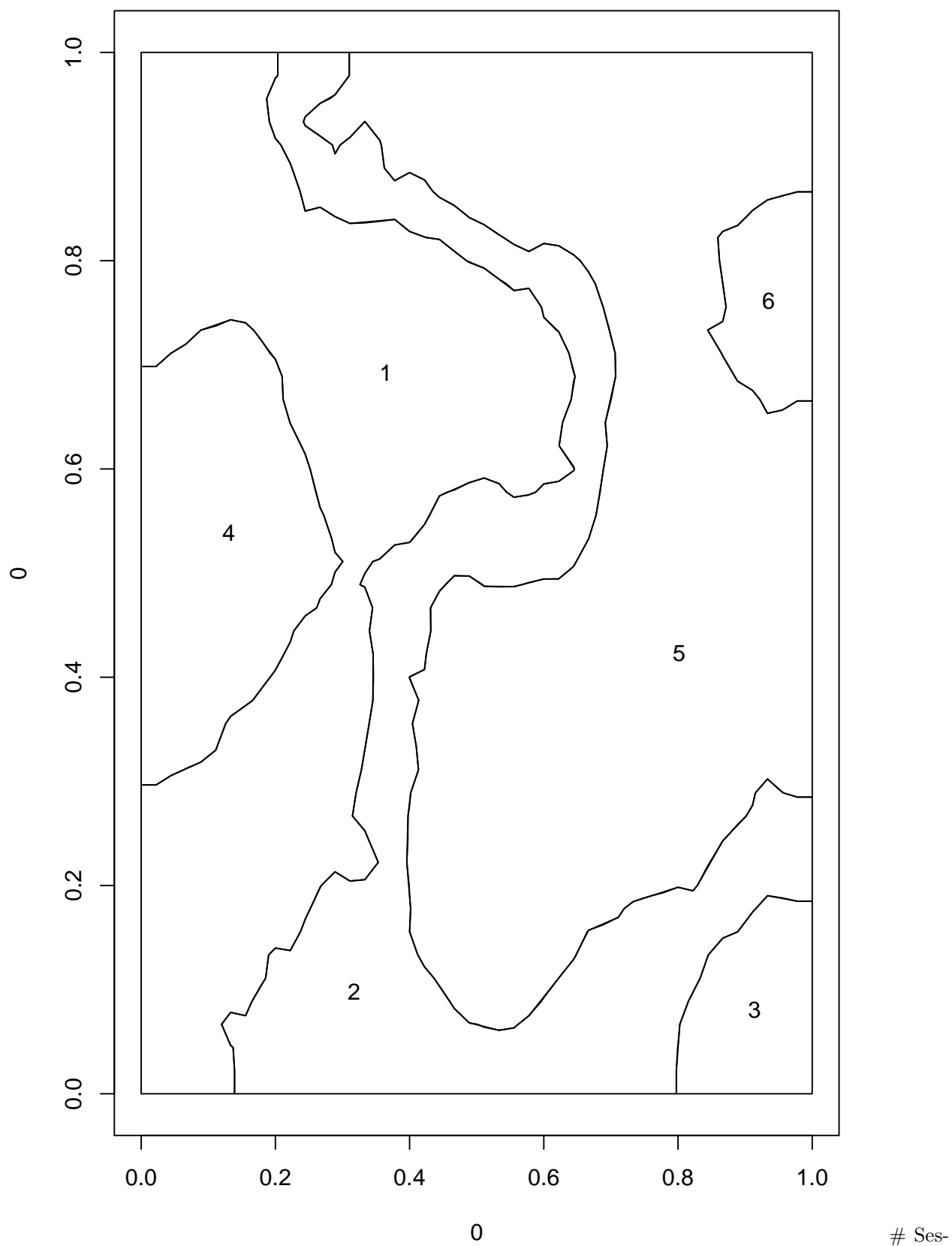
To keep only zones with a minimum of 20 data points, remake zoning structure.

```
K=calNei(Z,mapTest$krigData,mapTest$krigSurfVoronoi,mapTest$krigN,nmin=20)  
Z=K$zonePolygone  
plotZ(Z)
```

```
## NULL
```

```
title("Zoning with zones with less than 20 kriged data points removed")
```

Zoning with zones with less than 20 kriged data points removed



sion informations


```

## R version 3.4.0 (2017-04-21)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Debian GNU/Linux 8 (jessie)
##
## Matrix products: default
## BLAS: /usr/lib/libblas/libblas.so.3.0
## LAPACK: /usr/lib/lapack/liblapack.so.3.0
##
## locale:
##  [1] LC_CTYPE=fr_FR.utf8      LC_NUMERIC=C
##  [3] LC_TIME=fr_FR.utf8      LC_COLLATE=fr_FR.utf8
##  [5] LC_MONETARY=fr_FR.utf8  LC_MESSAGES=fr_FR.utf8
##  [7] LC_PAPER=fr_FR.utf8     LC_NAME=C
##  [9] LC_ADDRESS=C            LC_TELEPHONE=C
## [11] LC_MEASUREMENT=fr_FR.utf8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] grid      stats      graphics  grDevices utils      datasets  methods
## [8] base
##
## other attached packages:
## [1] gstat_1.1-5      fields_8.15      maps_3.1.1      spam_1.4-0
## [5] sp_1.2-4         rmarkdown_1.6    geozoning_1.0.0 devtools_1.13.3
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_0.12.11      compiler_3.4.0
##  [3] xts_0.9-7         tools_3.4.0
##  [5] digest_0.6.12     evaluate_0.10.1
##  [7] memoise_1.1.0     lattice_0.20-35
##  [9] rstudioapi_0.6    commonmark_1.2
## [11] yaml_2.1.14       knitr_1.17
## [13] withr_2.0.0       stringr_1.2.0
## [15] roxygen2_6.0.1    xml2_1.1.1
## [17] raster_2.5-8      desc_1.1.0
## [19] RandomFieldsUtils_0.3.25 rgeos_0.3-23
## [21] rprojroot_1.2     spacetime_1.2-0
## [23] R6_2.2.1          foreign_0.8-68
## [25] deldir_0.1-14     magrittr_1.5
## [27] htmltools_0.3.6   backports_1.1.0
## [29] intervals_0.15.1  RandomFields_3.1.50
## [31] maptools_0.9-2    assertthat_0.2.0
## [33] stringi_1.1.5     crayon_1.3.2
## [35] FNN_1.1           zoo_1.8-0

```