

TD : Objets flous spatialisés

Introduction

L'objectif de ce TD est de vous initier à la manipulation d'objets spatiaux flous, et ce en essayant de traiter un cas réel et compliqué, celui de l'identification des tronçons de route situés sur les rives de torrents. Dans un premier temps nous allons voir comment construire un objet spatial flou à partir de géométries vectorielles classiques, puis nous procéderons à des opérations entre objets flous avant d'aborder la question de la sélection floue.

Préparation de l'environnement de travail

La première étape consiste à supprimer tous les objets présents dans votre environnement de travail **R** afin d'éviter les effets de bords dans le cas où une précédente session est toujours active.

```
rm(list=ls())
```

Puis on charge les bibliothèques nécessaires à ce travail. Si ces dernières ne sont pas installées vous pouvez utiliser la fonction **install.packages()**

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

Vous pouvez ensuite configurer le répertoire de travail. Pensez à changer le dossier en fonction de votre poste de travail.

```
setwd("~/Cours/DESIGEO/ModèlesVagues/TD")
```

La préparation étant terminée nous pouvons aborder le TD à proprement parler.

Partie 1 : Création d'un objet flou

Pour commencer nous allons créer un objet spatial flou. Je vous propose de créer une représentation floue de torrents alpins. Le lit de ces derniers peut varier de manière significative entre l'hiver et le printemps, par conséquent il est difficile de fixer une limite nette à ces objets, doit-on se baser sur l'étendue minimale, maximale, moyenne, etc. De son côté l'ign à choisi de représenter ces objets avec différents polygones, certains représentant des surfaces d'eau permanentes et d'autres des surface d'eau intermittentes L'objectif de cette partie est de créer un objet flou à partir de ces données.

Mise en place

Importez la couche *Surface_eau.shp* présente dans le répertoire du TD à l'aide de la commande suivante :

```
surface_eau <- readOGR("./Data/Surface_eau.shp")
```

Vous définirez ensuite un raster vide dont les caractéristiques servirons de modèle pour la création des prochains rasters.

```
# Création d'un raster vide  
grid <- raster()  
# Son étendue est fixée à partir de l'étendue de la couche surface_eau  
extent(grid) <- extent(surface_eau)
```

```
# Sa résolution est fixée à 5m
res(grid) <- 5
```

1. À partir de la couche **surface_eau** créez deux nouvelles couches, la première composée uniquement des *surfaces d'eau permanentes* et une autre composée *uniquement des surfaces d'eau intermittentes*. Vous devez utiliser la syntaxe suivante en l'adaptant à votre requête.

```
nouvelle_table <- table[condition de sélection des lignes,]
```

2. Transformez ces couches en raster à l'aide de la fonction **rasterize()** en activant l'option **getCover** (c'est normal si c'est un peu lent).

```
# rasterisation
surface_eau_perm_r <- rasterize(surface_eau_perm, grid, getCover=TRUE)
surface_eau_inte_r <- rasterize(surface_eau_inte, grid, getCover=TRUE)
```

Cette opération étant couteuse en mémoire vous pouvez ensuite appeler le *garbage collector* afin de libérer un peu de mémoire pour la suite :

```
gc()
```

Nous allons maintenant transformer ces deux rasters en un objet spatial flou représentant les rivières. Pour ce faire nous allons procéder en deux étapes. Dans un premier temps nous allons définir une fonction permettant de calculer, pour les deux rasters construits, un degré d'appartenance à l'ensemble flous rivière, puis nous agrégerons ces deux raster pour construire l'objet final.

Création des sous-ensembles flous

Pour calculer le degré d'appartenance d'une cellule du raster à un ensemble flou il est nécessaire de définir une fonction d'appartenance, *i.e.* une fonction associant à chaque cellule une valeur entre 0 et 1. Lors de l'opération de rasterisation (cf. question 2) il était précisé d'activer l'option **getCover**, cette dernière active le calcul de la part de l'aire des cellules intersectée par la géométrie rasterisée. Cette dernière pouvant varier de 0 à 100. Je vous propose d'utiliser ce critère pour calculer le degré d'appartenance, avec la règle suivante: si plus de 75% de la superficie d'une cellule est couverte alors le degré d'appartenance est de 1, sinon il est de $x/75$ (avec x la part de l'aire couverte).

3. Implémentez la fonction décrite ci-dessus.

Une fois cette fonction créée on peut l'utiliser pour transformer les deux couches créées à la question 1 en objets flous.

4. Créez deux nouveaux rasters en appliquant aux raster précédents la fonction que vous venez de définir (à l'aide de la fonction **calc**).

Combinaison des sous-ensembles flous

Après avoir calculé deux objets flous, il est temps de les agréger pour construire l'objet flou **surface_eau**. Pour ce faire il va falloir écrire une nouvelle fonction calculant un degré d'appartenance en fonction du degré des deux rasters calculés lors de la question 4.

5. Définissez la fonction d'appartenance suivante. Quelle règle a été choisie ? Quelle est sa signification ?

```
fuzzy_combinaison <- function(x, y){
  val <- 0
  if(x[1] != 0 || y[1] != 0){
    val <- max(x/2,y)
  }
}
```

```

return(val)
}

```

6. Combinez les deux rasters créés à la question 4 à l'aide de la fonction que vous venez de définir (utilisez les fonctions **overlay()** et **Vectorize()**).

7. Exportez le raster nouvellement créé, importez le dans QGIS. Que pensez-vous du résultat ?

```

writeRaster(fuzzy_surface_eau_final, "fuzzy_surface_eau_final", overwrite=TRUE)

```

Si tout s'est bien passé vous venez de créer votre premier objet flou, félicitations.

Partie 2 : Construction de l'objet rives

Maintenant que nous avons construit une représentation floue des rivières, nous pouvons construire une représentation floue des rives de cette dernière. Le processus de construction est similaire, mais il présente une particularité, puisque il s'agit ici de construire un objet flou à partir d'un objet déjà flou, contrairement à la partie précédente.

Mise en place

8. Commencez par copier la couche créée lors de la question 7 (pour éviter d'écraser votre travail).

9. A l'aide de la fonction **distance()** calculez la distance entre les cellules dont le degré d'appartenance est nul et la cellule de degré d'appartenance non nul la plus proche. Pensez à lire la documentation de la fonction **distance()**, au risque de rater une étape.

10. Écrivez la fonction de fuzzification suivante. Quel est son fonctionnement ? Que représente l'objet flou **fuzzy_near** ?

```

fuzzy_near <- function(x){
  val <- 0
  if(x[1] < 100){
    val <- x[1] * -1/100 + 1
  }
  return(val)
}

```

11. Calculez, visualisez et commentez le raster **fuzzy_distance** définit ci-dessous.

```

fuzzy_distance <- calc(distance_surface_eau, fun=fuzzy_near)

```

12. Calculez l'objet rive à l'aide de la fonction suivante. Sachant que la fonction **min()** permet de calculer l'intersection de deux ensembles flous et que le complémentaire d'un ensemble flou de fonction d'appartenance y est 1-y, que déduisez-vous de cette fonction ? Comment est défini l'objet rives ?

```

fuzzy_imp <- function(x,y){
  val <- min(x[1], 1-y[1])
  return(val)
}

```

```

fuzzy_bank <- overlay(
  fuzzy_distance,
  fuzzy_surface_eau_final,
  fun=Vectorize(fuzzy_imp))

```

Partie 3: Sélection floue des routes

Dans cette dernière partie nous allons effectuer une sélection floue, *i.e* que nous allons attribuer à chaque objet un degré d'appartenance calculé en fonction de son degré de validation d'une règle

Encore une fois on commence par importer les données qui seront nécessaires pour ce travail.

```
Route <- readOGR("./Data/Route_segments.shp")
```

13. Écrivez une boucle permettant, pour chaque géométrie de la couche **Route**, de calculer le minimum de la valeur des cellules du raster **fuzzy_bank** intersectées par la géométrie (faites appel à la fonction **extract()**)
14. Ajoutez les données nouvellement calculées à la couche **Route** (à l'aide de la fonction **cbind()**)
15. Utilisez les fonction suivantes pour filtrer et exporter la couche, puis visualisez-la dans QGIS et commentez le résultat obtenu.

```
# On retire les Valeur 'Inf', i.e. les segments sortant du raster.
```

```
Route_filt <- Route[Route$out_val != Inf,]
```

```
writeOGR(Route_filt, "./out2.shp", "Route_filt", driver = "ESRI Shapefile")
```

Bonus (Dans QGIS)

1. Créez un buffer de 100m autour des surfaces d'eau intermittentes et permanentes
2. Sélectionnez tous les segments de route intersectant un buffer
3. Comparer le résultat de cette sélection avec le résultat de la partie 3, qu'en pensez-vous ?