



HAL
open science

TP manipulation et traitements de données LiDAR avec QGIS et R

Louis Rey

► **To cite this version:**

Louis Rey. TP manipulation et traitements de données LiDAR avec QGIS et R. Doctoral. Lyon 2, France. 2024.
<hal-05048934>

HAL Id: hal-05048934

<https://hal.science/hal-05048934v1>

Submitted on 28 Apr 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons CC BY 4.0 - Attribution - International License



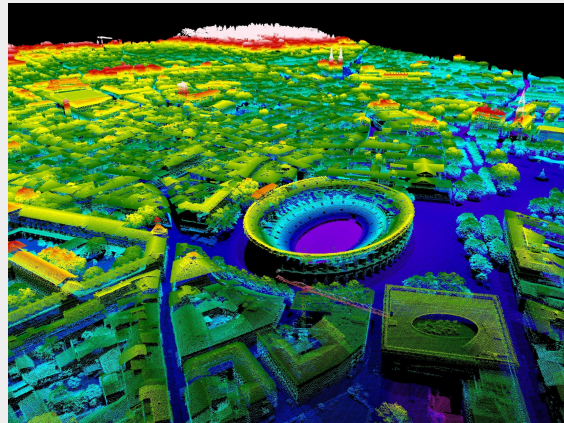
EVS

Environnement,
Ville, Société
UMR 5600



TP manipulation et traitements de données LiDAR avec QGIS et R

Octobre 2024



Source : [futura-sciences](https://www.futura-sciences.com)

Objectifs pédagogiques

- Manipuler des données LiDAR
- Utiliser les principaux algorithmes de classification des points sols
- Utiliser les principaux algorithmes d'interpolation raster
- Utiliser des outils gratuits et open-source R et QGIS pour le traitement des données LiDAR
- Faire une carte de lecture du paysage

Principaux logiciels gratuits pour manipuler des données LiDAR

1. QGIS et QGIS avec le package LasTools
2. CloudCompare
3. R avec package LidR

Contexte du TP

- Sortie terrain du 7 et 8 novembre 2024
- Sur la commune “Les Avenières Veyrins-Thuellin” en Isère
- A environ 4 km du Rhône actuel
- D’après la carte de l’état Major de 1836 il y avait des canaux dans cette zone

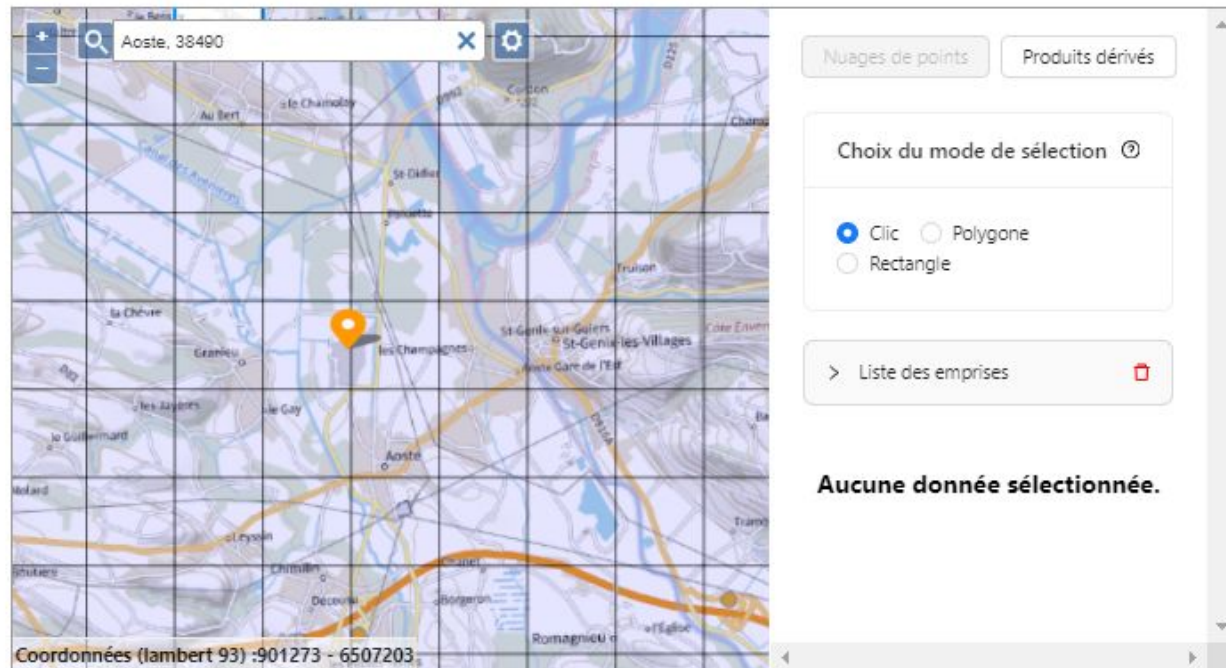
Problématique : **Peut-on identifier des paléochenaux du Rhône et peut-on identifier des anciens aménagements humains (ex : canaux de dérivation) ?**

Téléchargement d'une dalle de nuage LiDAR HD de l'IGN

Dalle nuage de points LiDAR HD classifié :

<https://geoservices.ign.fr/lidarhd#telechargementclassifiees>

Cliquez ici pour consulter la carte en plein écran.



The screenshot displays the IGN Lidar HD service interface. On the left, a map shows the area around Aoste, France, with a search bar at the top containing 'Aoste, 38490'. A red location pin is placed on the map. On the right, a control panel includes buttons for 'Nuages de points' and 'Produits dérivés', a 'Choix du mode de sélection' section with radio buttons for 'Clic' (selected), 'Polygone', and 'Rectangle', and a 'Liste des emprises' button. At the bottom of the panel, the text 'Aucune donnée sélectionnée.' is displayed. The map's coordinates are shown as 'Coordonnées (lambert 93) : 901273 - 6507203'.

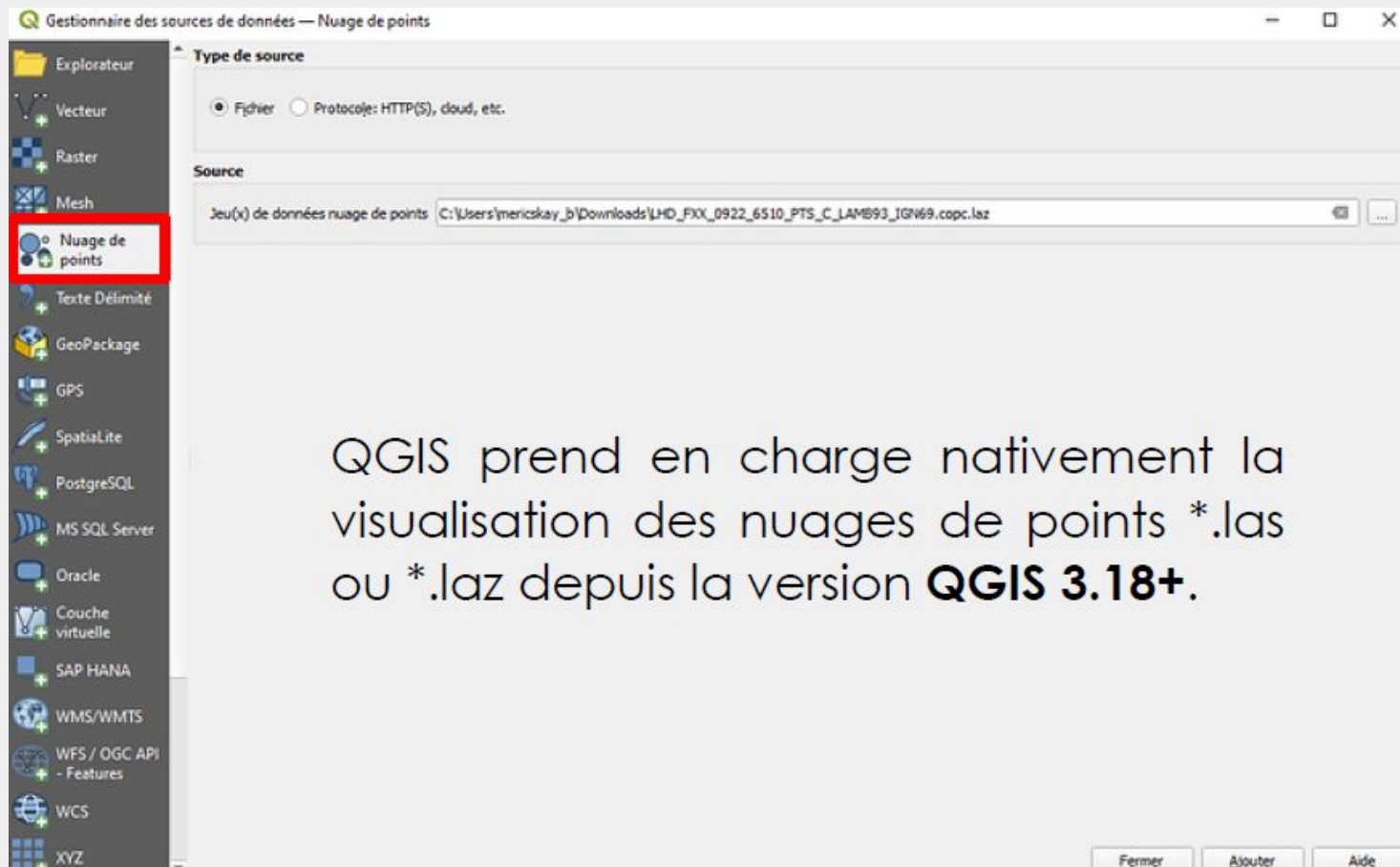
Téléchargement d'une dalle de nuage LiDAR HD de l'IGN

Depuis 2024, uniquement possible de télécharger les nuages de points LiDAR HD de l'IGN classifiées, ainsi pour ce TP vous devrez déclassifier votre nuage pour apprendre à le classifier

Dans QGIS il n'y a pas d'algorithme de classification des points gratuit pour le moment :

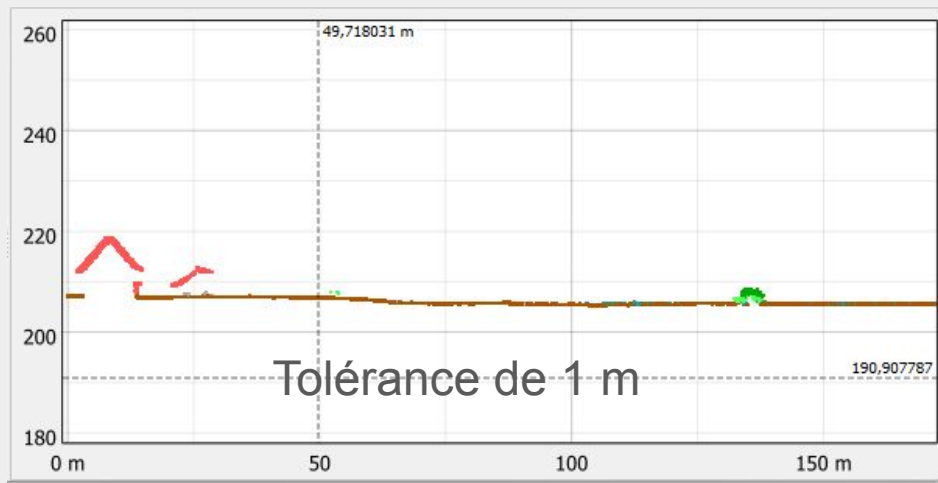
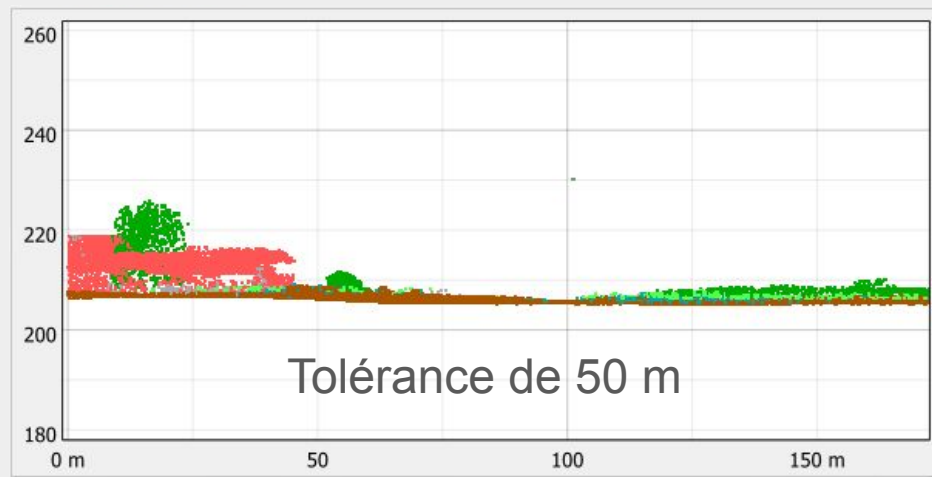
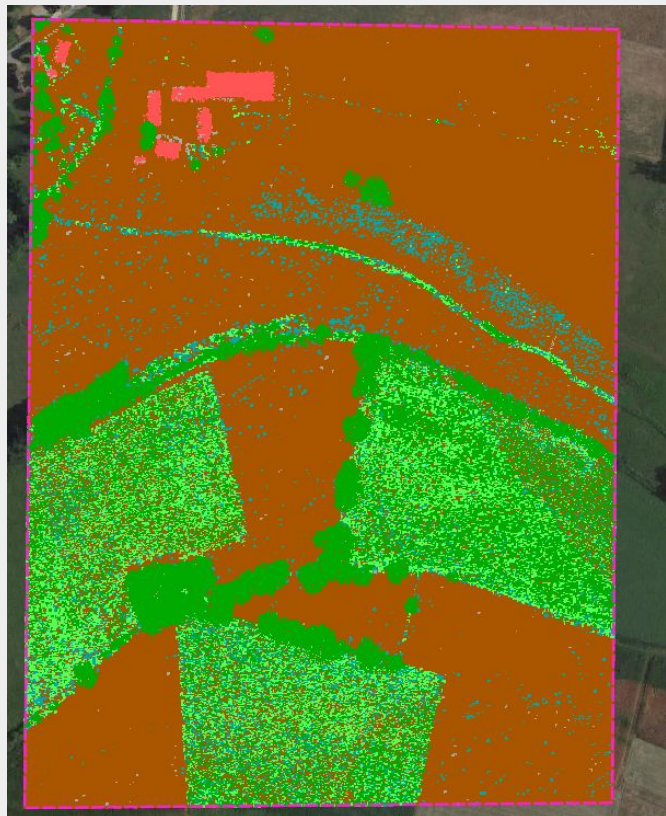
- Il est possible de passer par un package LasTools (payant à partir d'un certain nombre de données)
- Logiciel CloudCompare
- Langage de programmation (ex : R)

Téléchargement d'une dalle de nuage LiDAR HD de l'IGN



QGIS prend en charge nativement la visualisation des nuages de points *.las ou *.laz depuis la version **QGIS 3.18+**.

Utiliser le profil d'élévation de QGIS



Téléchargement d'une dalle de nuage LiDAR HD de l'IGN

Les formats de données LiDAR :

- **Le format *.LAS** est un format de fichier conçu pour l'échange et l'archivage de données de nuages de points LiDAR. Il s'agit d'un format binaire ouvert, spécifié par l'American Society for Photogrammetry and Remote Sensing (ASPRS).
- **Le format *.LAZ** est un fichier *.LAS compressé afin de faciliter son stockage et son partage. Les fichiers *.LAZ représentent généralement moins de 20 % de la taille du fichier *.LAS.
- **Les fichiers copc** : il s'agit d'un fichier *.laz indexé, ce qui signifie que l'indexation des points a été opérée afin de pouvoir y accéder plus rapidement. Il offre une structure de données arborescente où chaque noeud contient un sous-ensemble du nuage de points. Les index copc correspondent à des octrees, i.e des arbres avec systématiquement huit noeuds fils.

Classification de l'IGN

Les nuages de points LiDAR HD classifiés issus d'un processus très automatisé de classification internalisé (mélange algorithmes IA et classiques), mis au point par l'IGN (IGN,2024)

Attribut de classe du point :

- 8 classes respectant des valeurs prédéfinies des spécifications ASPRS ;
- 4 classes avec une valeur personnalisée (64, 65, 66 et 67).

| Valeur | Description |
|--------|-----------------------------------|
| 1 | Non classé |
| 2 | Sol |
| 3 | Végétation basse (0-50 cm) |
| 4 | Végétation moyenne (50 cm-1,50 m) |
| 5 | Végétation haute (+1,50 m) |
| 6 | Bâtiment |
| 9 | Eau |
| 17 | Tablier de pont |
| 64 | Sursol pérenne |
| 65 | Artefacts |
| 66 | Points virtuels |
| 67 | Divers - bâtis |

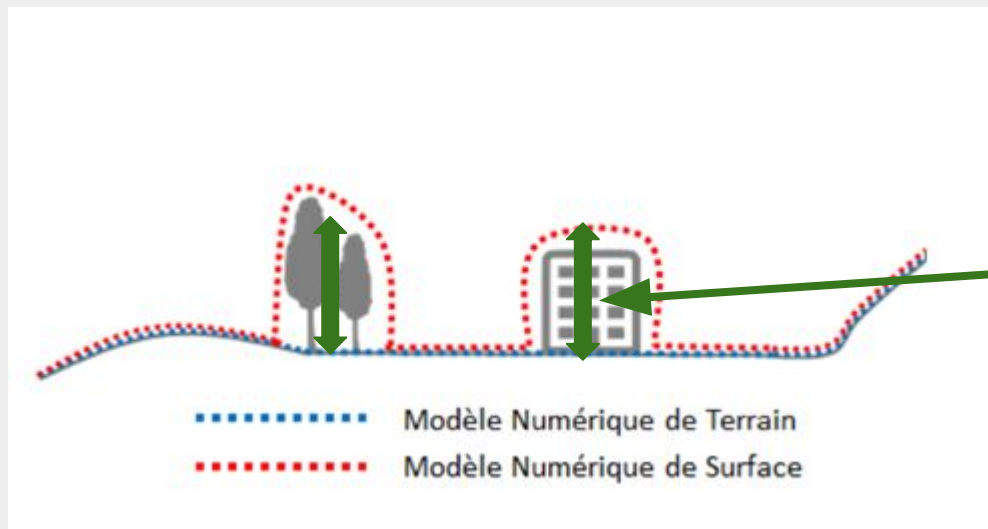
Observer votre nuage de points

1. Quelle est la date de création de cette donnée ?
2. Quelles sont les précisions altimétrique (z) et planimétrique (x,y) annoncées pour cette donnée? Est-ce que cela vous paraît réaliste?
3. Quel est le système de coordonnées planimétrique de cette donnée
4. Quel est le système de coordonnées altimétrique de cette donnée?

Principaux paramètres

| Attribut au sein des données | Point Data Record Format 6 (standard ASPRS) | Traduction |
|------------------------------|---|-----------------------------|
| x | X | x |
| y | Y | y |
| z | Z | z |
| intensity | Intensity | Intensité du signal retour |
| return | Return Number | Numéro de l'écho |
| number_of_return | Number of Returns (given pulse) | Nombre d'écho |
| classification_flags | Classification Flags | Caractéristique du point |
| scanner_channel | Scanner Channel | Canal du scanner |
| scan_direction | Scan Direction Flag | Sens de balayage du scanner |
| flight_line_edge | Edge of flight line | Fin d'axe de vol |
| classification | Classification | Classe |
| user_data | User Data | Donnée utilisateur |
| angle | Scan Angle | Angle de scan |
| point_source_id | Point Source ID | ID de l'axe de vol |
| gps_time | GPS Time | Temps absolu |

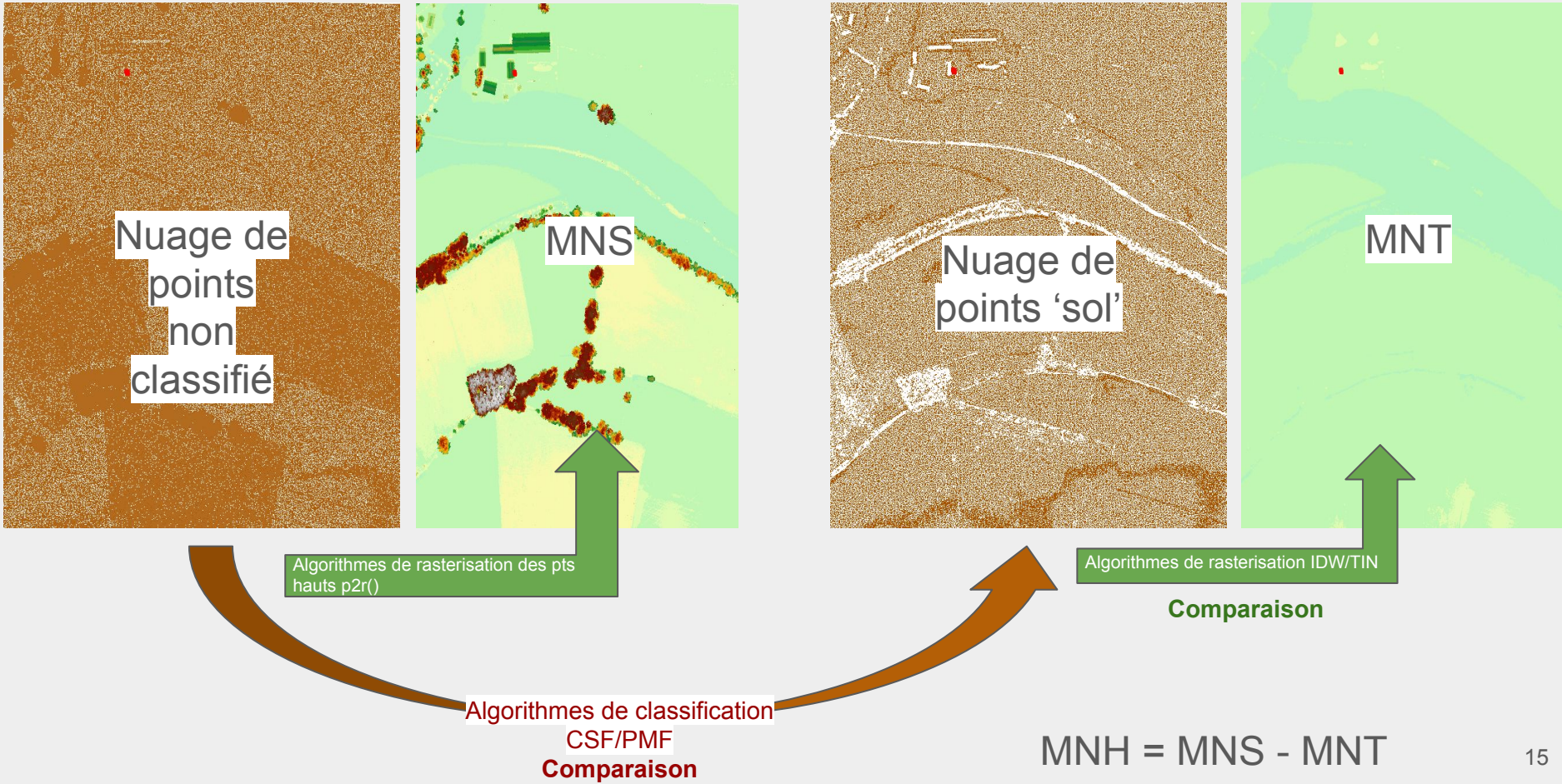
Délivrables courantes avec un LiDAR



Modèle Numérique de Hauteur
(MNH) = MNS - MNT

Source : [geoservices 2024](#)

Chaînes de traitements du TP



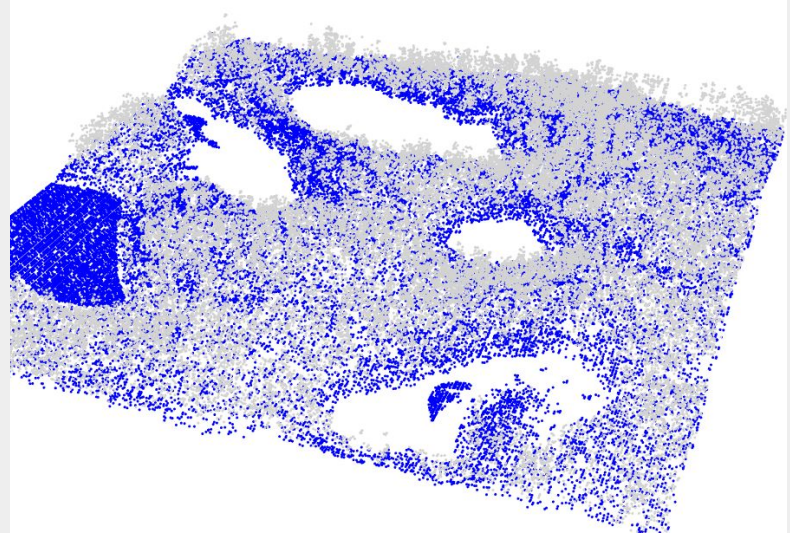
An aerial view of a city, likely Paris, with a large circular stadium (Stade de France) in the center. The image is overlaid with a colorful point cloud or heatmap, where colors range from blue (low elevation) to red (high elevation). The stadium is a prominent feature, surrounded by a dense urban area. The text "Comment obtenir un MNT ?" is overlaid on the image in a white box.

Comment obtenir un
MNT ?

Classifier les points sols selon deux méthodes courantes

CSF : Cloth Simulation Filter

PMF: Progressive Morphological Filter



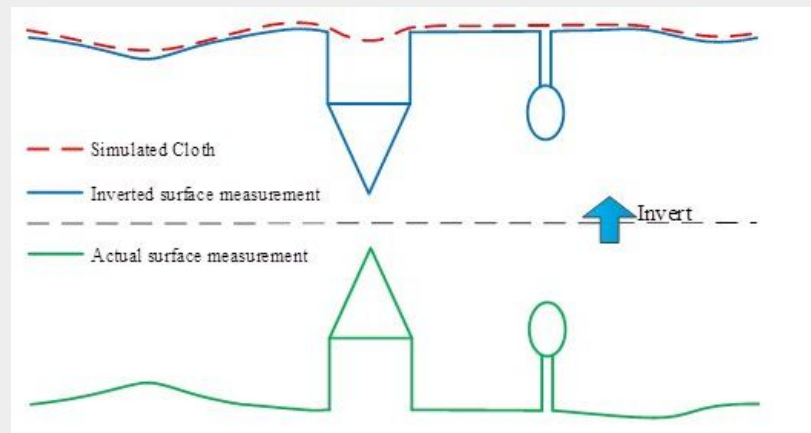
Source : [LidR](#)

Classification des points sol → Objectif : MNT



CSF : Cloth Simulation Filter

est une approche physique inspirée par une simulation de la chute d'un drap ou d'une toile (cloth). Article scientifique de Wuming Zhang, Jianbo Qi, Peng Wan, Hongtao Wang (School of Geography, Beijing Normal University, China)



Source : [Wikipédia, 2024](#)

| Parameter | Value | | |
|--------------------------|--------------------|-------------------|--------|
| Terrain relief amplitude | sharply undulating | gently undulating | flat |
| Cloth rigidity | 1 | 2 | 3 |
| Spring-back ratio | 30% | 51% | 65.70% |

Classification des points sol → Objectif : MNT



CSF : Cloth Simulation Filter

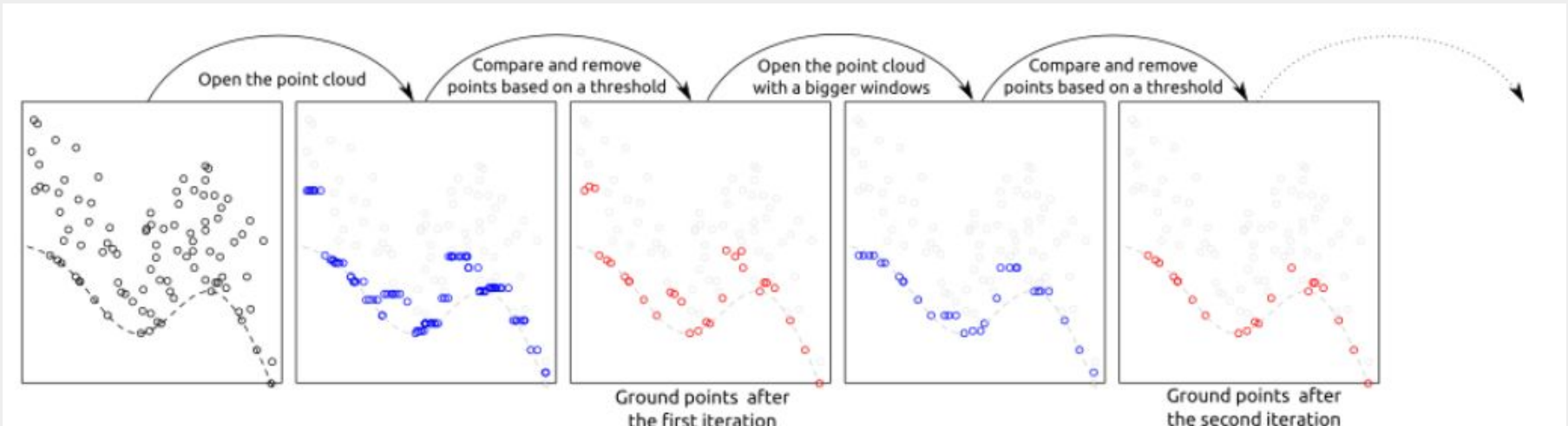
| Paramètres | Explications | Standard (par défaut) |
|-------------------------|---|------------------------|
| cloth_resolution | ajustable en fonction de la granularité désirée. | 1 (unité projection) |
| max_iterations | à chaque itération, il classe progressivement davantage de points comme étant au sol, en se basant sur des critères géométriques.(mais ++ de temps) | 500 |
| class_threshold | ajustable selon la hauteur des objets hors-sol à détecter. | 0.5 (unité projection) |
| rigidness | dans des zones très accidentées, vous pouvez réduire la rigidité pour que le tissu suive mieux le relief. | 3 |
| time_step | Il définit le pas de temps utilisé dans la simulation de la chute du tissu. Plus la valeur est petite, plus le tissu tombe lentement et peut s'adapter aux petites variations du sol. | 0.65 |

Classification des points sol → Objectif : MNT



PMF: Progressive Morphological Filter → itératif

Filtre progressivement les points à partir des fenêtres de taille croissante pour identifier ceux qui appartiennent au sol.



Source : [LidRbook](#)

Classification des points sol → Objectif : MNT



PMF: Progressive Morphological Filter → itératif

| Paramètres | Explications | Standard (par défaut) |
|------------|------------------------------------|-----------------------|
| ws | window size | À vous de choisir |
| th | threshold height for ground points | À vous de choisir |

Source : [LidRbook](#)

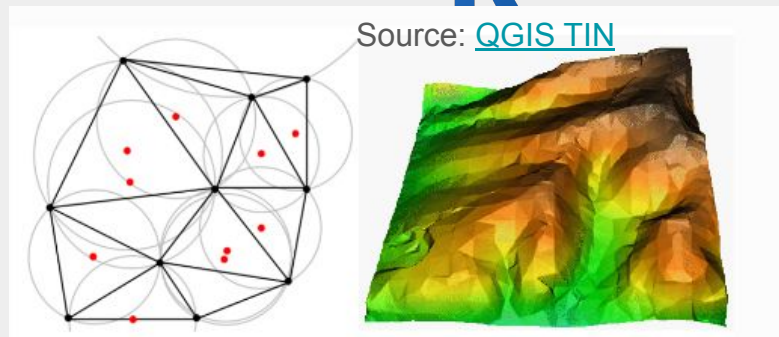
Avantages vs inconvénients CSF et PMF

| Méthode | Fonctionnement | Avantage(s) | Inconvénient(s) |
|------------|---|--|---|
| CSF | La méthode CSF simule un drap suspendu au-dessus d'un nuage de points LiDAR, qui s'ajuste progressivement pour s'étendre et "tomber" sur les points les plus bas, identifiant ainsi les points du sol. | +++ zones complexes et denses ++ Fonctionne bien pour des topographies irrégulières | -Temps de calcul pouvant être conséquent si gros nuage de points - Peut nécessiter un paramétrage complexe dans certains cas |
| PMF | La méthode PMF implémente un algorithme de segmentation de points du sol basé sur un filtre morphologique progressif. Paramètre Th = Limite pour hauteur max sol et Ws = contrôle la taille de la fenêtre locale | ++ Efficace en zone peu complexe (ex : zone urbaine) +++Rapide | -Fonctionne mal en environnement complexe comme des berges de rivières, forêts... |

Rasterisation des points sol (algorithme TIN)



TIN (Triangulated Irregular Network)



Tente de créer une surface de triangles formés par les points voisins les plus proches.

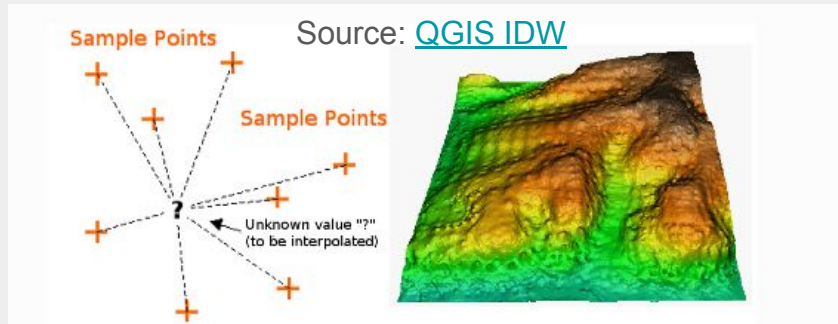
Le principal inconvénient de l'interpolation TIN est que les surfaces ne sont pas lisses et peuvent donner un aspect irrégulier. Cela est dû à des pentes discontinues sur les bords du triangle et des points de données d'échantillons

Dans R fonction `rasterize_terrain()`. L'option "method" te permet de choisir entre TIN et IWD

Rasterisation des points sol (algorithme IDW)



IDW (Inverse Distance Weighting) est une méthode d'interpolation qui repose sur l'idée que les points proches d'un point à estimer ont plus d'influence que les points éloignés.



L'altitude de ce point est déterminée par une moyenne pondérée des altitudes des points voisins. Plus un point est proche de celui à estimer, plus son influence sera importante, car le poids attribué diminue avec la distance.

Dans R “algorithm = knnidw(k=10)”

Knnidw : K-Nearest Neighbors (KNN) et Inverse Distance Weighting (IDW).

Avantages inconvénients TIN et IWD

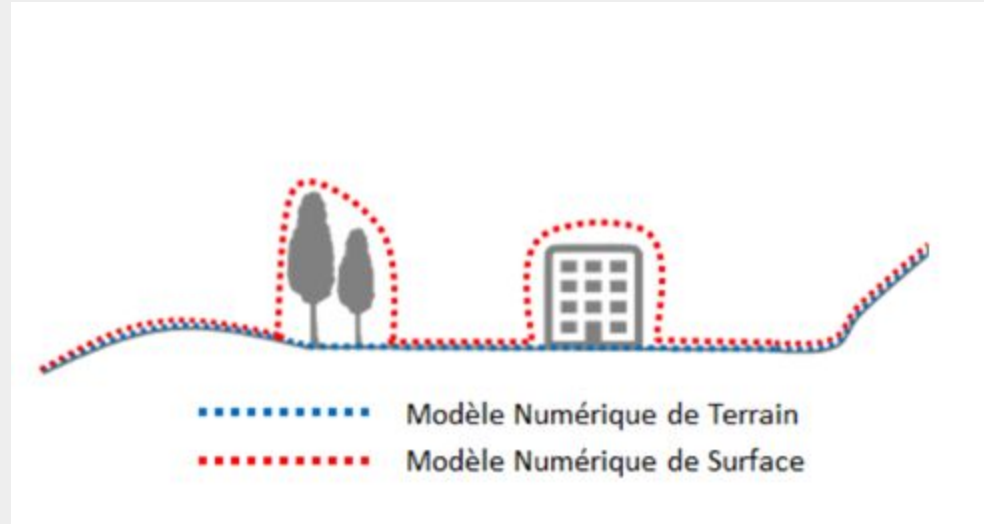


| Méthode | Avantage | Inconvénients |
|---------|---|---|
| TIN | Exploite pleinement la structure irrégulière des points | Peut donner des zones du MNT avec beaucoup de petits sommets |
| IWD | Fonctionne bien pour des terrains relativement réguliers et lorsque les points sont uniformément répartis | Lisse les zones et peut être embêtant pour des zones de crêtes ou de changement topographique |

An aerial view of a large, circular stadium, likely the Stade de France, with a colorful point cloud overlay. The stadium's interior is highlighted in red, and the surrounding area, including trees and buildings, is rendered in various colors like green, blue, and yellow. A white text box is overlaid on the stadium.

Comment obtenir un
MNS ?

Classification des points les plus hauts → Objectif : MNS



Source : [geoservices 2024](#)

Rasterisation des points les plus hauts → Objectif : MNS

`rasterize_canopy ()` : Cette fonction du package `lidR` est utilisée pour générer un raster de hauteur de la canopée. Hauteur la plus haute

`p2r()` : C'est un algorithme de rasterisation (point-to-raster) utilisé avec `rasterize_canopy()` (source : [R_GIS_04](#))

Option `Subcircle` = l'algorithme ne prend en compte que les points situés dans un cercle de rayon. **Une fois le point le plus haut trouvé pour chaque cellule, le raster est généré avec la hauteur maximale**

Exemple : `mns_LiDAR_IGN<- rasterize_canopy(nuageHDclasse, res = 0.45, algorithm = p2r(subcircle = 0.15))`

An aerial photograph of a village with a prominent church spire, overlaid with a color-coded elevation map. The map uses a color scale from blue (low elevation) to red (high elevation). The church spire is a prominent feature, colored in a dark red. The surrounding landscape includes fields, trees, and buildings. The text "Quelle résolution spatiale choisir ?" is overlaid on the image in a white box.

Quelle résolution spatiale
choisir ?

Quelle résolution spatiale choisir ? (exemple)



Ex : 20 points/m²

Quelle taille de pixel faut il choisir pour avoir au moins **1 point/pixel** ?

Quelle résolution spatiale choisir ? (exemple)



Ex : 20 points/m²

Quelle résolution [m] de pixel faut il choisir pour avoir au moins **1 point/pixel** ?

Densité : Nb pts/surface = 20pts/m²

Spixel [m²] : Stotale/Nb pts

Résolution [m] : $\sqrt{(\text{Spixel})}$

Quelle résolution spatiale choisir ? (exemple)



Ex : 20 points/m²

Quelle résolution [m] de pixel faut il choisir pour avoir au moins **1 point/pixel** ?

22,36 cm

Avant de commencer le TP

- Ouvrir R et RStudio et QGIS (version >3.18)
- Créer vous un projet QGIS là où vous avez enregistré le dossier avec les données et le diaporama du TP
- Créez vous un projet RStudio et enregistré là où vous avez enregistré le dossier également
- Créer un nouveau script R et enregistrez le au même endroit

Traitements

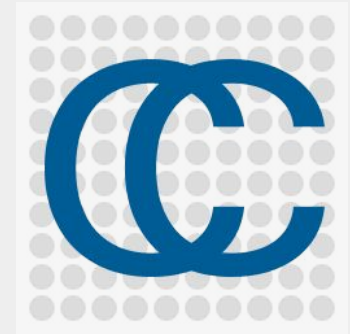


+

Visualisation



Optionnel



Court rapport

Un rapport comprenant :

- (i) Une explication succincte des principales lignes de code
- (ii) Une réponse aux questions avec des captures d'écran si besoin



Préparation de l'environnement R studio



Ouvrir R studio

Créer un nouveau projet à l'endroit où vous avez copié le dossier Tp_LiDAR que je vous ai transmis

Créer un nouveau script et sauvegarder le après l'avoir nommé

Pour savoir où sont enregistrés vos packages, taper : `>> .libPaths()`

Assurez vous d'avoir une seule version de R installée sur votre ordinateur, risque de problèmes de dépendance autrement (aller dans C:\Program Files\R)

Connaître la version R que vous avez tapé dans la console : `>> R.version.string`

Préparation de l'environnement R studio



#installer les packages

```
install.packages("ggplot2")
```

```
install.packages("lidR")
```

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

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

activer les librairies

```
library(ggplot2)
```

```
library(lidR)
```

```
library(raster)
```

```
library(sp)
```

Charger le fichier .laz de l'IGN



```
nuageHDclasse <- readLAS("nuage_coupe.laz")
```

```
print(nuageHDclasse)
```

```
plot(nuageHDclasse)
```

```
nuageHDclasseSol <- filter_poi(nuageHDclasse, Classification == 2)
```

```
print(nuageHDclasseSol)
```

```
plot(nuageHDclasseSol)
```

```
writeLAS(nuageHDclasseSol , "nuage_classifieIGN_sol_secteur_1.laz")
```



Déclassification nuage de points

Regarder les différentes classes dans votre nuage :

```
nuage@data$classification *
```


Reclassifier les points :

```
nuageHDclasse @data$Classification <- 0
```

```
plot(nuageHDclasse , color = "Z", size = 1)
```

```
writeLAS(nuageHDclasse , "nuage_brut.laz")
```

```
nuageHDdeclasse <-readLAS("nuage_brut.laz")
```



| Valeur | Description |
|--------|-----------------------------------|
| 1 | Non classé |
| 2 | Sol |
| 3 | Végétation basse (0-50 cm) |
| 4 | Végétation moyenne (50 cm-1,50 m) |
| 5 | Végétation haute (+1,50 m) |
| 6 | Bâtiment |
| 9 | Eau |
| 17 | Tablier de pont |
| 64 | Sursol pérenne |
| 65 | Artefacts |
| 66 | Points virtuels |
| 67 | Divers - bâtis |

*Le symbole @ est utilisé pour accéder à un slot d'un objet S4. S3 où on utilise le symbole \$ pour les dataframes

Classifier les points sols selon les deux méthodes principales



Cloth Simulation Filter (CSF) #package lidR

```
nuage_classif_csf <- classify_ground(nuage, csf()) # csf est la méthode la plus  
courante de classification des pts sols
```

```
print(nuage_classif_csf)
```

```
plot (nuage_classif_csf, color ="Classification")
```

#Filtrer uniquement pts sol

```
nuage_classif_csf_sol <- filter_poi(nuage_classif_csf, Classification ==2)
```

```
#package lidR
```

```
plot(nuage_classif_csf_sol)
```

```
writeLAS (nuage_classif_csf_sol, "nuage_classif_csf_sol.laz")
```

Filtrer uniquement les points sols

Progressive Morphological Filter (PMF)

```
nuage_classif_pmf <- classify_ground(nuage, algorithm = pmf(ws = 5, th = 0.2))
```

```
print(nuage_classif_pmf)
```

```
plot (nuage_classif_pmf, color ="Classification")
```

#Filtrer uniquement pts sol

```
nuage_classif_pmf_sol <- filter_poi(nuage_classif_pmf, Classification ==2) # lidR
```

```
plot(nuage_classif_pmf_sol)
```

```
writeLAS (nuage_classif_pmf_sol, "nuage_classif_pmf_sol.laz")
```

Question 1 : Quelles différences entre les nuages de points sol CSF et PMF et lequel semble le mieux approprié pour votre terrain d'étude ?

Filtrer le nuage classifié de l'IGN pour ne garder uniquement les points sols 

```
plot(nuageHDclasse)
```

```
nuageHDclasse_sol <- filter_poi(nuageHDclasse, Classification==2)
```

```
plot(nuageHDclasse_sol)
```

```
writeLAS (nuageHDclasse_sol, "nuageHDclasse_sol.laz")
```

Question 2 : Comparer la densité des différents nuages de points sol : (1) IGN classifié (2) Classifié par CSF (3) Classifié PMF. Qu'observez vous ? Comparer les nuages de points ?

Rasteriser les points sols en faisant une interpolation (MNT)



Méthode TIN Triangulated Irregular Network #par défaut résolution 1 mètre

#CSF et TIN

```
Mnt_csf_tin <- rasterize_terrain(nuage_classif_csf_sol, algorithm = tin(), res=0.25)
writeRaster(Mnt_csf_tin, "mnt_csf_tin.tif", overwrite = TRUE)
```

#PMF et TIN

```
mnt_pmf_tin <- rasterize_terrain(nuage_classif_pmf_sol, algorithm = tin(), res=0.25)
writeRaster(mnt_pmf_tin, "mnt_pmf_tin.tif", overwrite = TRUE)
```

Rasteriser les points sols en faisant une interpolation (MNT)

Méthode IDW Inverse Distance Weighting

Paramètre k : Cela indique que l'algorithme utilisera les k plus proches voisins pour estimer la valeur des cellules du raster

#CSF et IDW

```
mnt_csf_idw <- rasterize_terrain(nuage_classif_csf_sol, algorithm = knnidw(k=10), res = 0.25)
```

```
writeRaster(mnt_csf_idw, "mnt_csf_idw.tif", overwrite = TRUE)
```

#PMF et IDW

```
mnt_pmf_idw <- rasterize_terrain(nuage_classif_pmf_sol, algorithm = knnidw(k=10), res = 0.25)
```

```
writeRaster(mnt_pmf_idw, "mnt_pmf_idw.tif", overwrite = TRUE)
```

Question 3 : Comparer les MNT interpolation IDW et interpolation TIN. Laquelle des interpolations vous semble la plus adaptée ? et pourquoi ?

Rasteriser les points sol du nuage **déjà classifié** de l'IGN (MNT)

```
print(nuageHDclasse_sol)
```

#IDW

```
mnt_lidarHD_idw<-rasterize_terrain(nuageHDclasse_sol, algorithm =  
knnidw(k=10),res=0.25)
```

```
writeRaster(mnt_lidarHD_idw, "mnt_lidarHD_idw.tif", overwrite = TRUE)
```

Question 4 : Comparer le MNT à partir des points sols CSF et interpo IDW avec le MNT points sol classifiés par l'IGN interpo IDW. Qu'observez vous ? Lequel semble meilleur ?

Rasteriser les points de canopé (MNS)

```
mns_LiDAR_IGN<- rasterize_canopy(nuageHDclasse, res = 0.45, algorithm =  
p2r(subcircle = 0.15))
```

```
plot(mns_LiDAR_IGN)
```

```
writeRaster(mns_LiDAR_IGN, "mns_LiDAR_IGN.tif")
```



Calculer le MNH

```
res(mns_LiDAR_IGN)  
res(mnt_lidarHD_idw)
```

```
crs(mns_LiDAR_IGN)  
crs(mnt_lidarHD_idw)
```

```
print(mns_LiDAR_IGN)  
print(mnt_lidarHD_idw)
```

```
mnt_lidarHD_idw_resampled <- resample(mnt_lidarHD_idw, mns_LiDAR_IGN)
```

```
mnh<- mns_LiDAR_IGN - mnt_lidarHD_idw_resampled
```

```
writeRaster(mnh , "mnh.tif")
```

Question 5 : Quels intérêts d'utiliser un MNH ?

Carte

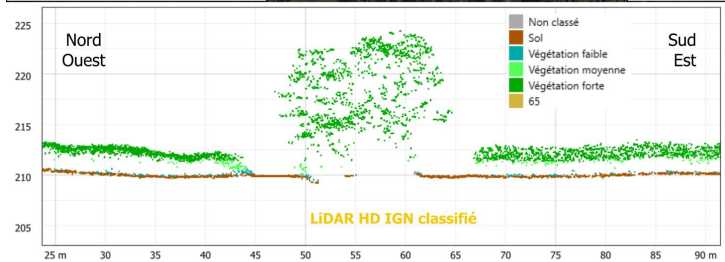
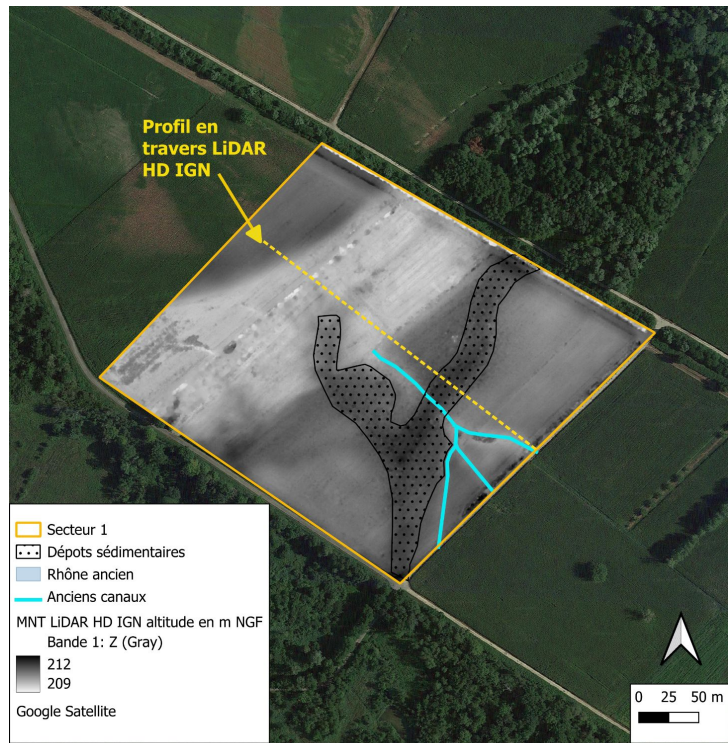
Doit comprendre :

- (i) Une carte avec une vue 2D du MNT LiDAR HD IGN avec vos analyses de paysages
- (ii) Un profil en long montrant une coupe du nuage de points brute (prendre un maximum de diversité du paysage)

Vous pouvez vous aider du MNT- 1m de l'IGN

Remarques :

- Doit apparaître la légende, la flèche du Nord, l'échelle et toutes autres informations qui vous semblent utiles.
- Pour le choix de la zone de profil, choisissez une zone pertinente.
- Utilisez de préférence QGIS → vous pouvez utiliser d'autres outils pour améliorer votre carte finale, libre à vous



Merci pour votre écoute !

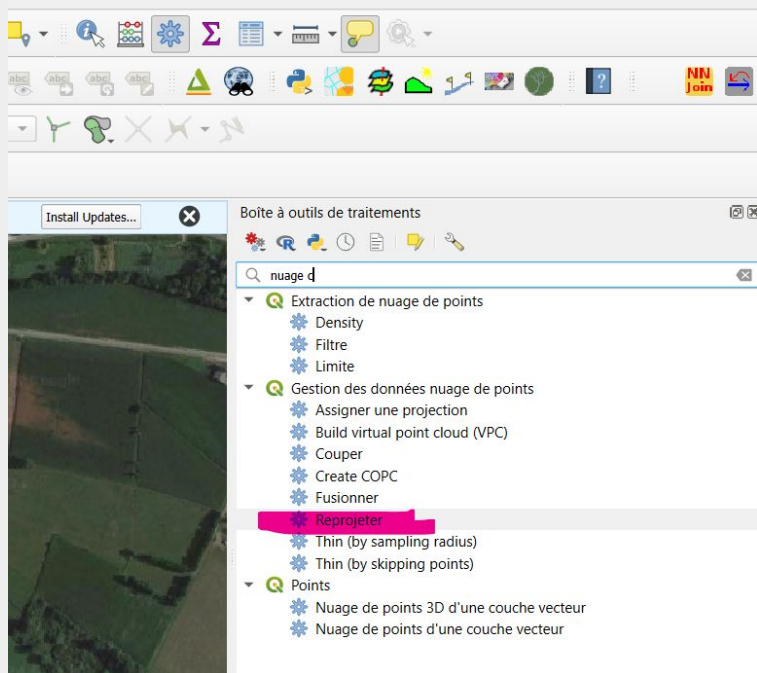
Avez vous des questions ?

Vous pourriez en avoir besoin



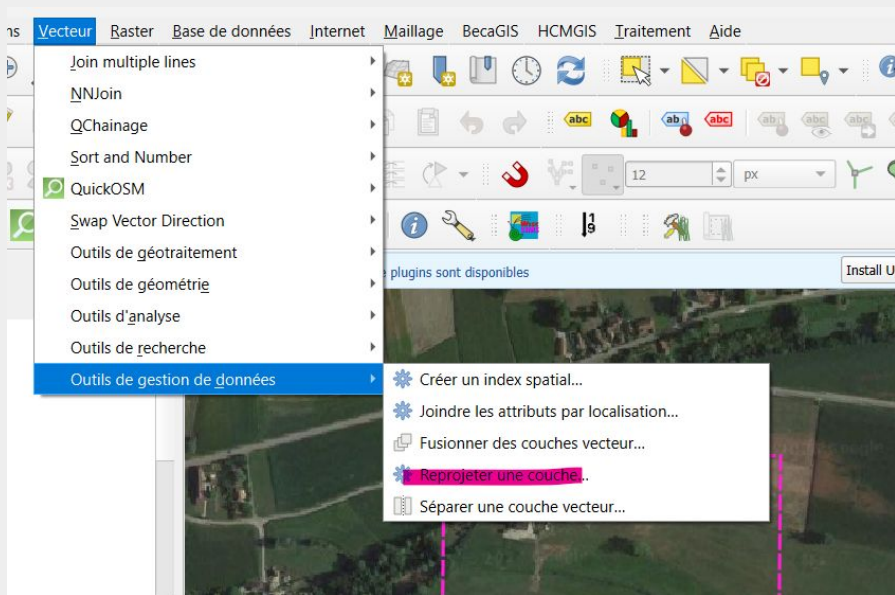
Reprojeter le nuage de points

```
las <- lastransform(las, epsg_to = 4326)
```



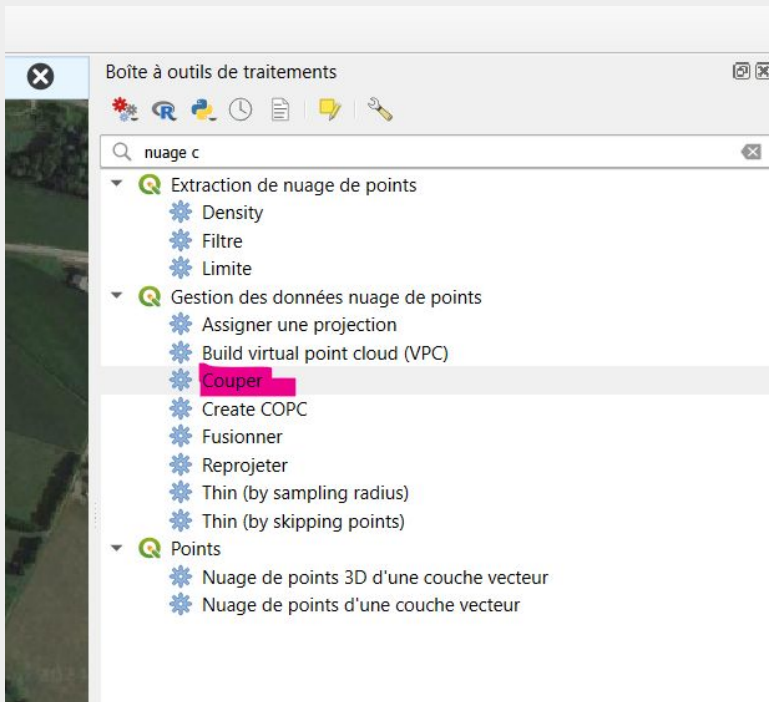
Reprojeter le vecteur d'emprise

```
emprise <- st_transform(emprise, crs = EPSG0)
```



Couper un LiDAR par vecteur

```
lasCoupe <- lasclip(las, emprise)
```



Fusionner des nuages de points LiDAR avec



```
folder_path <- getwd() # récupérer chemin dans lequel votre projet est enregistré
```

```
nomFichiers_laz <- list.files(path = folder_path, pattern = "\\*.laz$", full.names = T) # $ signifie la fin de la chaîne
```

```
las_list <- lapply(las_files, readLAS) # lapply permet d'appliquer une fonction à chaque élément d'une liste
```

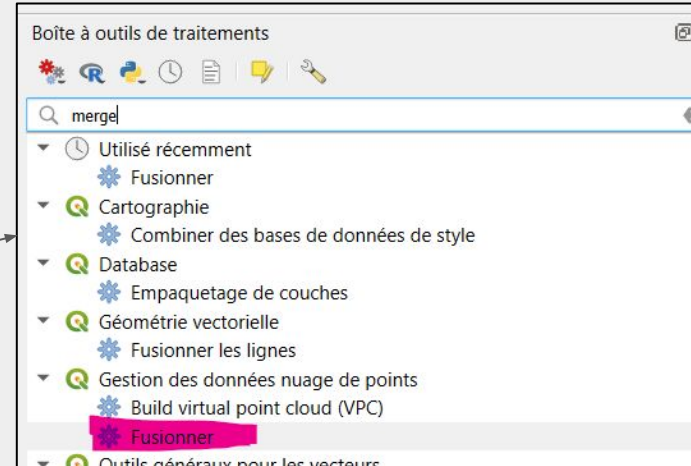
```
nuage_fusionne <- do.call(rbind, las_list)
```

```
writeLAS(nuage_fusionne, "nuage_fusionne.laz")
```

Fusionner des nuages de points LiDAR avec



Gestion des données nuage de points > Fusionner



Avantages vs inconvénients des méthodes d'interpolation raster

| Méthode | Fonctionnement | Avantage(s) | Inconvénient(s) |
|---|---|---|---|
| Triangulated Irregular Network (TIN) | La triangulation de Delaunay est unique et l'interpolation linéaire est sans paramètre. | la solution la plus simple car elle ne fait intervenir aucun paramètre | créer un MNT non lisse et qu'elle ne peut pas extrapoler le terrain en dehors de l'enveloppe convexe délimitée par les points au sol car il n'y a pas de facettes de triangle en dehors de l'enveloppe convexe. |
| Invert distance weighting (IDW) | Elle repose sur l'hypothèse selon laquelle la valeur à un point non échantillonné peut être estimée comme une moyenne pondérée des valeurs à des points situés à une certaine distance limite d ou à partir d'un nombre k donné de voisins les plus proches. | elle est plus résistante aux artefacts de bord car elle utilise un voisinage plus pertinent | génère des terrains « bosselés » et probablement pas aussi réalistes que ceux générés à l'aide de TIN. |
| Krigeage | Est une méthode d'interpolation géostatistique qui estime la valeur d'une variable en un point inconnu en se basant sur les valeurs observées des points voisins, tout en prenant en compte la structure spatiale et la covariance entre ces points. Il permet ainsi de minimiser l'erreur d'estimation en intégrant des modèles de variogramme pour décrire la variabilité spatiale. | approche la plus avancée et la plus fiable | Complexe et prend du temps à être utilisé avec fiabilité |

Manipulation CloudCompare (optionnel)

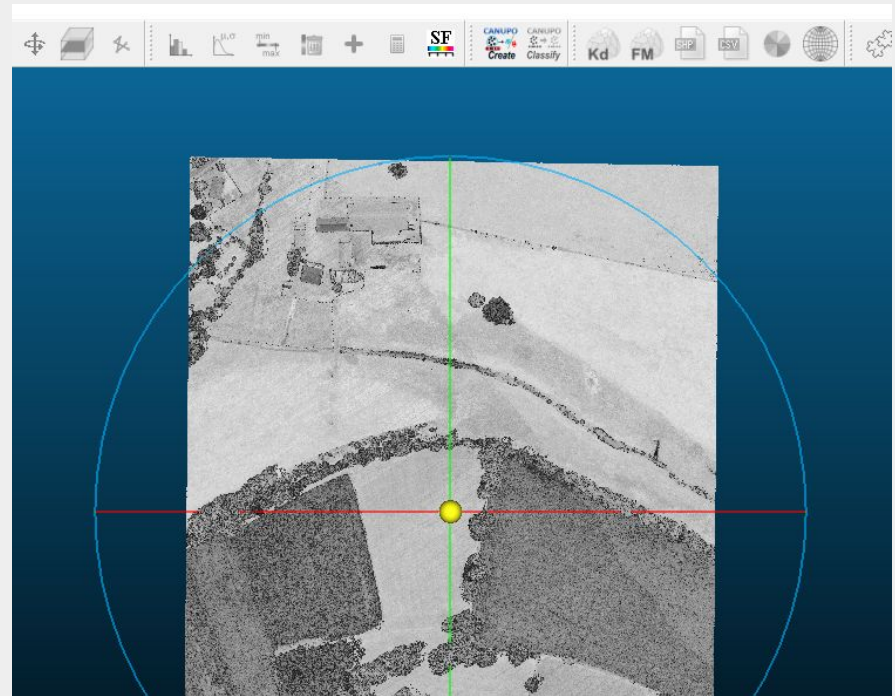


CloudCompare est un logiciel Open Source Français permettant la manipulation, l'affichage, la comparaison et le recalage de nuage de points provenant de données laser et Photogrammétrie.

Ouverture du nuage de points 3D lidar dans CloudCompare

File>Open ou glisser et déposer le nuage classifié de l'IGN

Du fait de la longueur des coordonnées X Y, CloudCompare va appliquer une translation au fichier pour diminuer le poids du fichier



Global shift/scale

shift/scale information is stored and used to restore the original coordinates at export time

| | | | |
|---|----------------|---|--|
| Point in original coordinate system (on disk) x = 895571.380000 y = 6510350.190000 z = 207.250000 | + Shift | Previous input -895764.46 -6510656.02 0.00 | Point in local coordinate system x = -193.08000 y = -305.83000 z = 207.25000 |
| | x Scale | 1.00000000 | |

Preserve global shift on save

Yes Yes to All No

Visualiser le nuage de points

Regarder les propriétés

: click gauche sur le nuage dans l'onglet DB Tree

Dans l'onglet Scalar Field sélectionner "classification"

Régler les différents paramètres : colors scale et SF display params

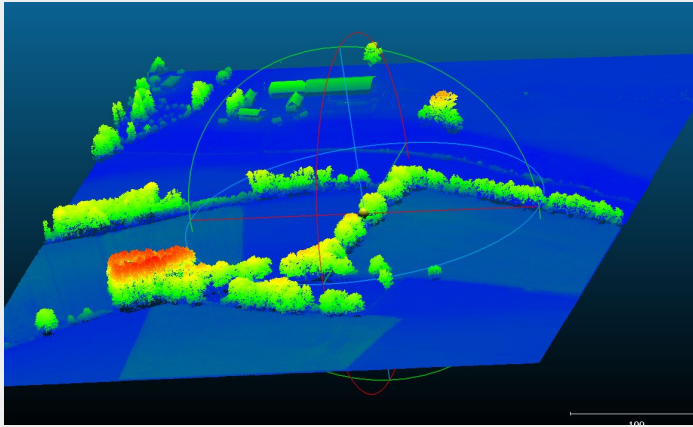
The screenshot displays the CloudCompare v2.13.1 interface. The DB Tree on the left shows a folder named 'nuage_coupe.copc.laz' containing a file 'nuage_coupe.copc.laz'. The Properties panel on the right shows the following details for the selected object:

| Property | State/Value |
|-------------------|---|
| CC Object | |
| Name | nuage_coupe.copc.laz |
| Visible | <input checked="" type="checkbox"/> |
| Colors | SF Scalar field |
| Show name (i...) | <input type="checkbox"/> |
| Box dimensions | X: 462.11 (-231.05 : 231.06) Y: 612.68 (-306.33 : 306.35) Z: 38.9 (204.12 : 243.02) |
| Shifted box ce... | X: 0.00499725 Y: 0.0100098 Z: 223.57 |
| Global box ce... | X: 895764.464997 Y: 6510656.030010 Z: 223.570007 |
| Info | Object ID: 269 - Children: 0 |
| Current Display | 3D View 1 |
| Cloud | |
| Points | 6,940,761 |
| Global shift | (-895764.46;-6510656.02;0.00) |
| Global scale | 1.000000 |
| Point size | Default |
| Scalar Fields | |
| Count | 7 |

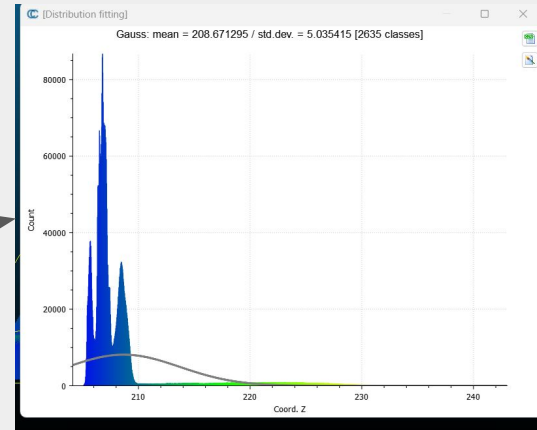
The 3D View window on the right shows a point cloud visualization of a landscape. The ground is colored in shades of blue and green, while buildings and trees are highlighted in yellow and orange. A red crosshair is visible in the center of the view.

Le visualiser avec une couleur selon l'altitude et regarder la distribution des valeurs

Tools>Projection>>Export coordinates to SF



Dans la barre du haut sélectionner l'histogramme

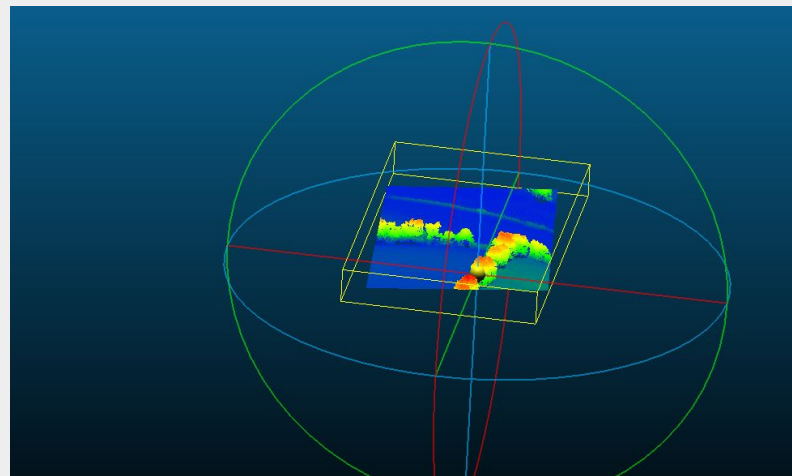
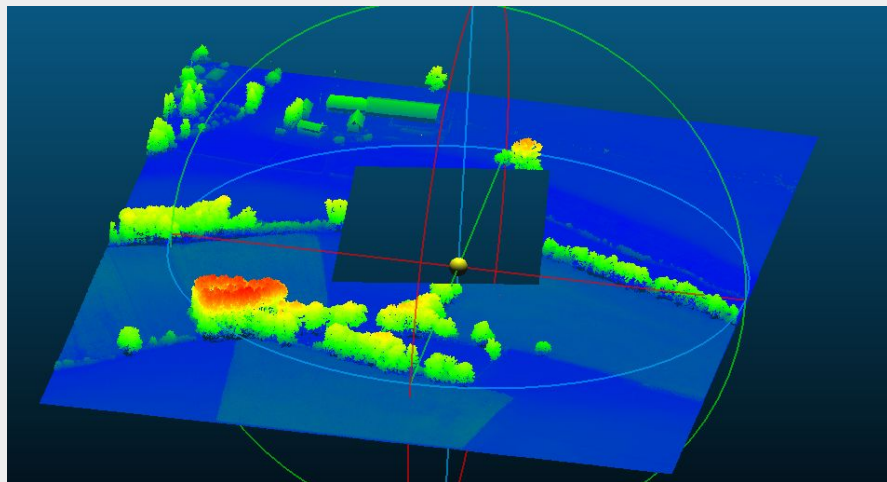


Segmenter manuellement votre nuage de points

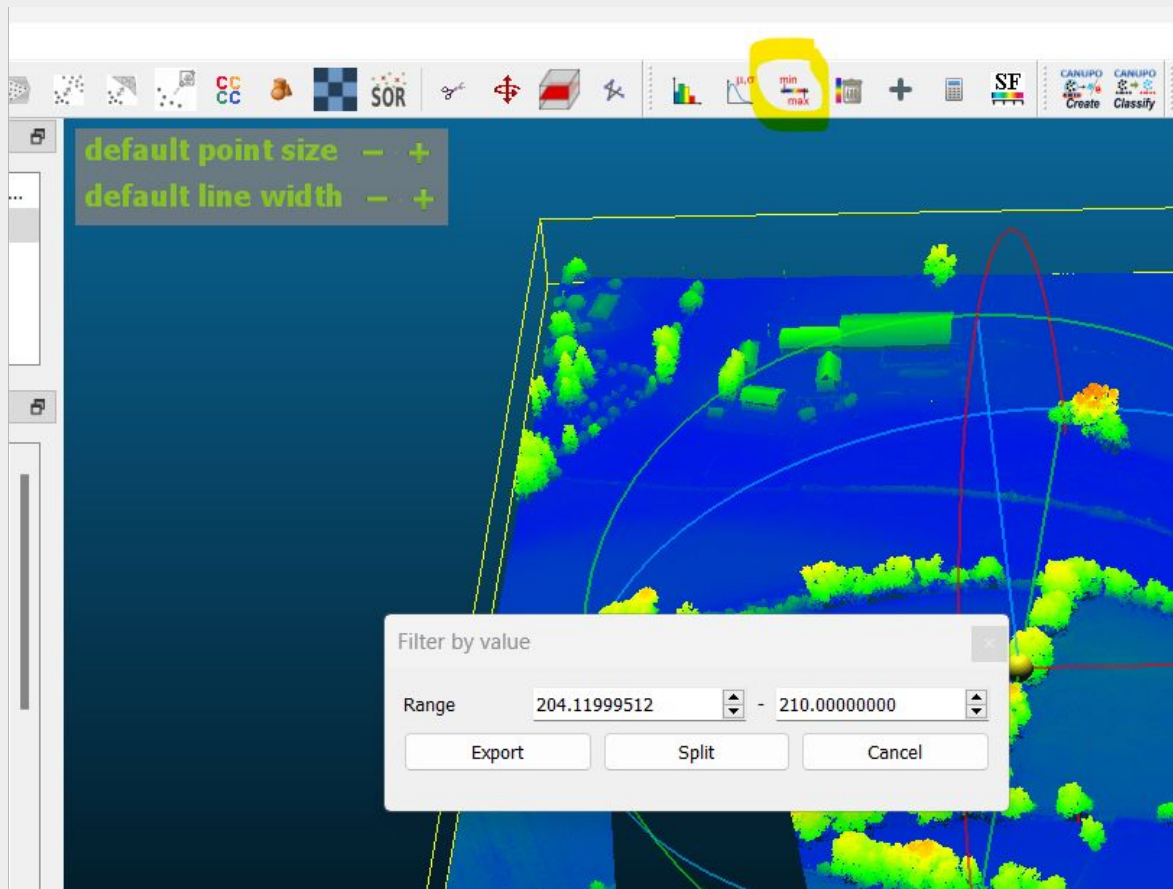
Outil ciseaux dans la barre du haut



Une nouvelle barre apparaît avec des outils pour le ciseaux

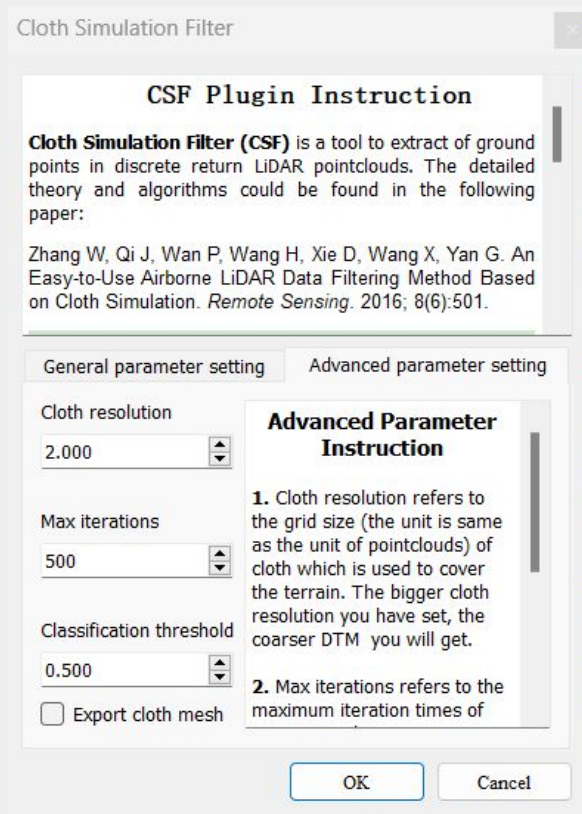
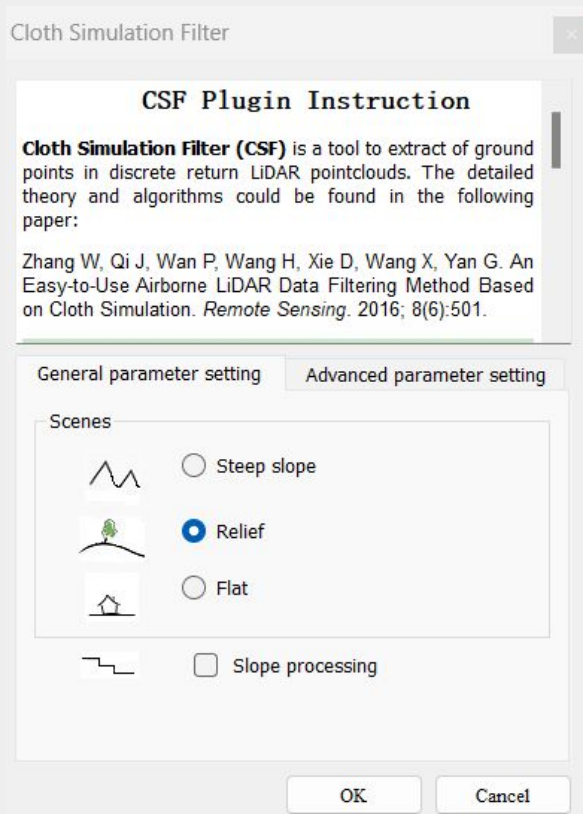


Segmenter à partir d'un intervalle



Classifier le sol avec l'algorithme "CSF"

Onglet CSF
filter (barre
d'affichage de
gauche)



Bibliographie

Blog SIG & Territoires:

<https://www.sigterritoires.fr/index.php/qgis-pour-lidarmodele-numerique-de-terrain-mnt-avec-lastools/>

LidRbook : <https://r-lidar.github.io/lidRbook/>

Exploiter les nuages de points LIDAR avec qgis 3.34:

<https://www.youtube.com/watch?v=NBkkf-H1G0Y&t=448s>

Vidéo youtube pour faire carte QGIS 3D, 2D et profil elevation :

<https://www.youtube.com/watch?v=Ocbb86cn8VE>

CloudCompare Wikipédia : <https://fr.wikipedia.org/wiki/CloudCompare>

Installation et récupération des données LiDAR

Installer l'extension Open LiDAR Toolbox, Lastool

Doit être installer les binaires/exécutables pour le système d'exploitation respectif :

<https://rapidlasso.de/downloads/> (Lastool)

Installer sur le C:\ les
extension

Explication GitHub pour Lastool →

<https://github.com/stefaneichert/OpenLidarToolbox?tab=readme-ov-file#classify-las-laz>