

Cartographie des routes forestières au Québec et mise en production à grande échelle avec lidR

- Jean-Romain Roussel
- Post-doctorant à Université Laval, Québec
- Développeur du package R lidR



Sujets de la présentation

Présentation faite de 2 présentations semi-indépendantes sur:

1. Cartographie automatique des routes forestières (5-6 min)

- Problématique canadienne même s'il existe des travaux français sur le sujet
- Généralité sur la méthode sans rentrer dans les détails



2. Utilisation du package lidR (10-12 min)

- Généralités sur les capacités du package
- Généralités sur le design du package
- En prenant l'exemple de la cartographie des routes forestières



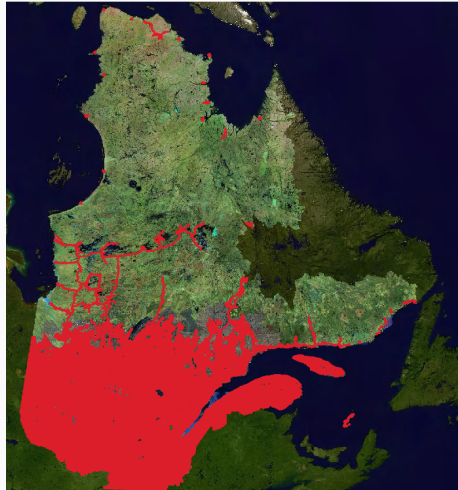
Cartographie des routes

Problématique

Quantité gigantesque et éloignement important des routes.

Besoin d'outils automatiques pour maintenir à jour les cartes.

Il existe déjà des travaux de recherche de cartographie des routes à l'international



Les travaux sont basés sur de la classification de rasters.

Problèmes:

- Ce sont des ensembles de pixels disjoints
- Sans topologie intrinsèque
- Volumineux en mémoire mais contiennent peu d'information



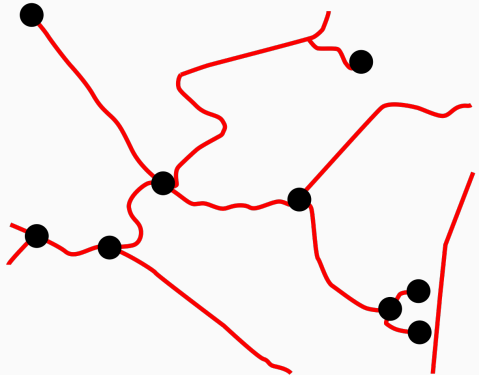
Ce ne sont pas des réseaux et on ne peut pas les utiliser.

Problématique

Les travaux sont basés sur de la classification de rasters.

Problèmes:

- Ce sont des ensembles de pixels disjoints
- Sans topologie intrinsèque
- Volumineux en mémoire mais contiennent peu d'information



Ce ne sont pas des réseaux et on ne peut pas les utiliser.

La problématique est donc la suivante:

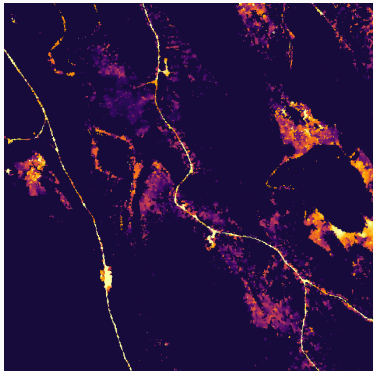
*Segmenter les routes forestières et produire une carte **vectorielle et topologiquement valide** c'est à dire un vrai réseau vectoriel de routes connectées*

Solution

La méthode consiste à **conduire** numériquement une carte de probabilités générée par une méthode de classification/segmentation



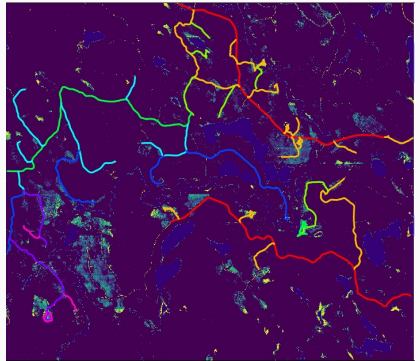
Kearney et al 2020, RGB, CNN



Roussel et al 2022, lidar, procedural

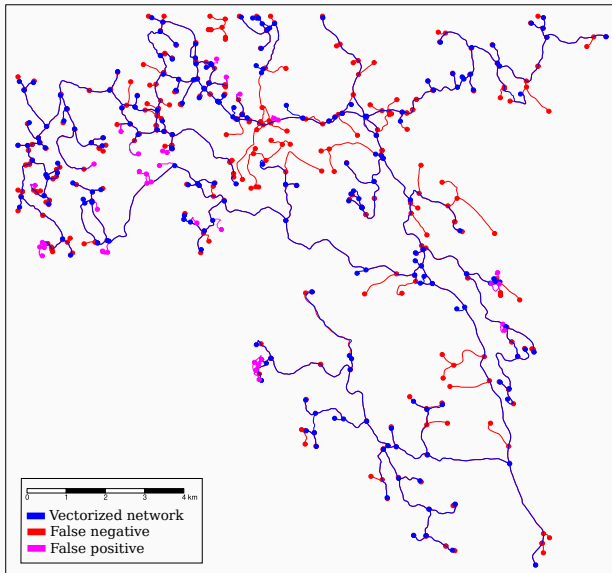
Conduire numériquement est
conceptuellement trivial

- Le produit est nativement vectoriel
- La topologie se construit automatiquement



Voir gif animé

Solution

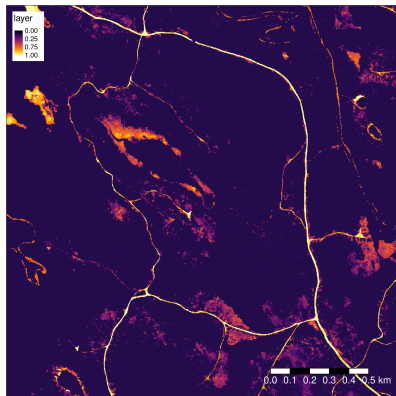


Carte de probabilités

Le but est de “faire ressortir” les routes

La carte contient de nombreux de faux positifs et faux négatifs

Conduire sur la carte permet de naturellement d’être robuste aux FP et FN.



Le package lidR

Idée générale

La plupart des utilisateurs utilisent lidR pour faire des CHM, DTM, ITD, ITS etc.

On pourrait faire la même chose avec FUSION ou LAStools. Ce que lidR offre c'est quelque de plus.

J'aimerais vous faire sentir ce que j'ai en tête lorsque je développe lidR et vous donner une idée de ce que vous pouvez en faire.

Carte de probabilités de route

```
probability_map = function(point_cloud, ...) {  
  # do something complex with "point_cloud"  
  return(map)  
}
```

C'est du nouveau code, jamais imaginé par personne d'autre et ça pourrait être **votre** code, **votre** idée innovante.

Entrée: un petit nuage de points en mémoire.

Sortie: un petit raster de probabilités.

Et maintenant ?

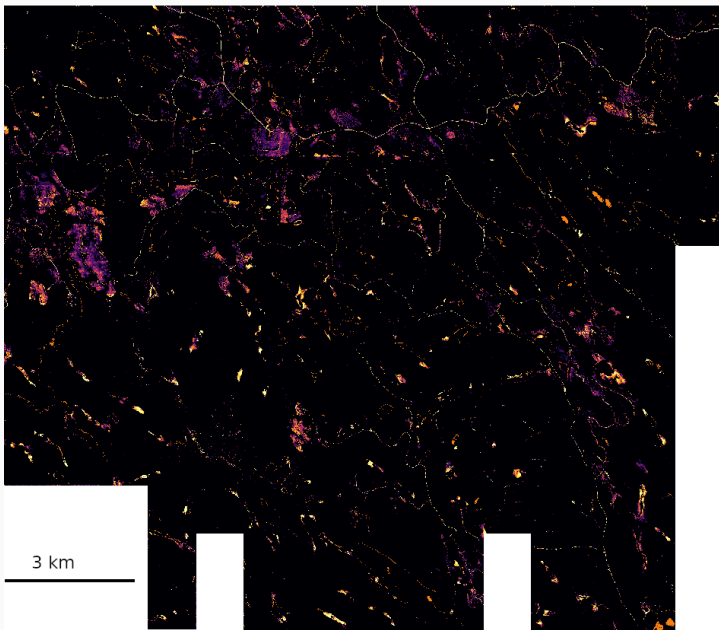
On veut appliquer cette fonction à toutes les tuiles sur 1000 km²

- En parallèle
- Sur les premiers retours uniquement
- Sans surcharger la mémoire
- **En chargeant un buffer de points pour ne pas avoir d'effet de bord**
- En gérant les erreurs potentielles
- En suivant la progression en temps réel

C'est possible de le coder mais c'est **très lourd**. C'est pourquoi lidR contient déjà un moteur de traitement générique et versatile.

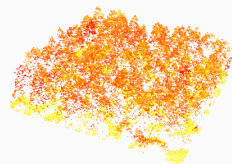
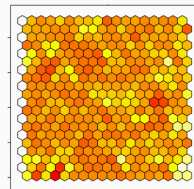
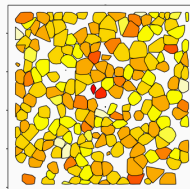
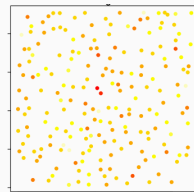
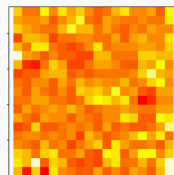
```
pmap = catalog_map(collection , probability_map, ...)
```

Et c'est tout !



lidR est une boîte à outils

- Extraction de points
(requête spatiale,
spatialement indexée)
- Lissage des points
- Détection de géométries
(lignes, plans)
- Filtrage de points
- **Calcul versatile de
métriques à de nombreux
niveaux de régularisation**
- ...



lidR est une boîte à outils

Avec `pixel_metrics()`

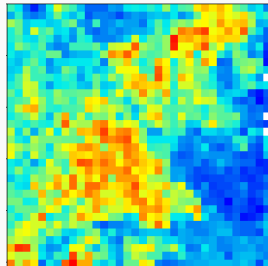
- Modèles prédictifs ABA
- Modèle de canopée
- Segmentation de lac (raster)
- **Carte de probabilités de route**
- ...

Avec `tree_metrics()`

- Modèle prédictif par arbre
- Classification de végétation
- Classification d'essence
- ...

Avec `point_metrics()`

- Colorisation de données multispectrales
- Lissage d'intensité
- Filtrage d'outliers



Je découvre moi même des utilisations possibles régulièrement

lidR est une boîte à outils

Avec `pixel_metrics()`

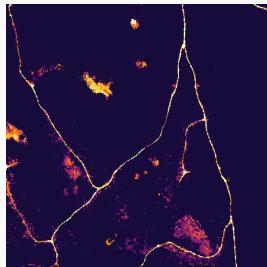
- Modèles prédictifs ABA
- Modèle de canopée
- Segmentation de lac (raster)
- **Carte de probabilités de route**
- ...

Avec `tree_metrics()`

- Modèle prédictif par arbre
- Classification de végétation
- Classification d'essence
- ...

Avec `point_metrics()`

- Colorisation de données multispectrales
- Lissage d'intensité
- Filtrage d'outliers



Je découvre moi même des utilisations possibles régulièrement

lidR est une boîte à outils

Avec `pixel_metrics()`

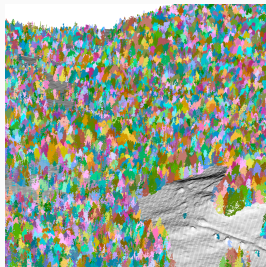
- Modèles prédictifs ABA
- Modèle de canopée
- Segmentation de lac (raster)
- **Carte de probabilités de route**
- ...

Avec `tree_metrics()`

- Modèle prédictif par arbre
- Classification de végétation
- Classification d'essence
- ...

Avec `point_metrics()`

- Colorisation de données multispectrales
- Lissage d'intensité
- Filtrage d'outliers



Je découvre moi même des utilisations possibles régulièrement

lidR est une boîte à outils

Avec `pixel_metrics()`

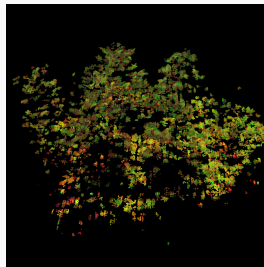
- Modèles prédictifs ABA
- Modèle de canopée
- Segmentation de lac (raster)
- **Carte de probabilités de route**
- ...

Avec `tree_metrics()`

- Modèle prédictif par arbre
- Classification de végétation
- Classification d'essence
- ...

Avec `point_metrics()`

- Colorisation de données multispectrales
- Lissage d'intensité
- Filtrage d'outliers



Je découvre moi même des utilisations possibles régulièrement

lidR est une boîte à outils

Avec `pixel_metrics()`

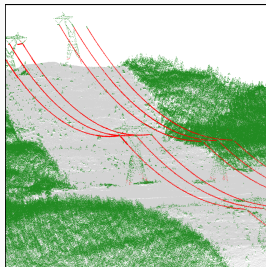
- Modèles prédictifs ABA
- Modèle de canopée
- Segmentation de lac (raster)
- **Carte de probabilités de route**
- ...

Avec `tree_metrics()`

- Modèle prédictif par arbre
- Classification de végétation
- Classification d'essence
- ...

Avec `point_metrics()`

- Colorisation de données multispectrales
- Lissage d'intensité
- Filtrage d'outliers



Je découvre moi même des utilisations possibles régulièrement

lidR supporte les plugins

Votre code

```
novel_sementation = function(point_cloud, p1, p2) {  
  # do something complex with "point_cloud"  
  return(segmented_trees)  
}
```

Intégration dans lidR par plugin

```
novel_sementation = plugin_its(novel_sementation)  
segment_tree(collection, novel_sementation(0.5, 3))
```

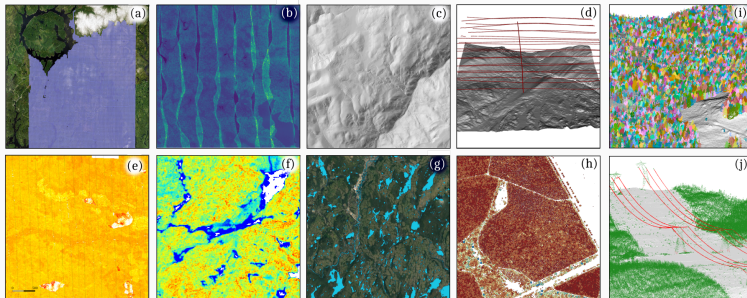
2 lignes pour passer d'un algorithme qui fonctionne localement à un algorithme applicable à large échelle.

C'est comme ça que je développe de nouvelles choses. En (presque) 2 lignes de codes je passe d'une idée testée localement à un outil prêt à la production.

But: produire de la recherche reproductible.

Conclusion sur la conception sous-jacente de lidR

- lidR n'as pas été conçu pour faire des CHM, DTM ou classifier des points sol dans un contexte opérationnel.
- lidR à été conçu pour permettre de construire facilement de nouvelles méthodes inédites et les mettre en production dans un contexte académique.
- C'est en cela que lidR est différent de FUSION ou LAStools (ou autres).



Merci pour votre attention