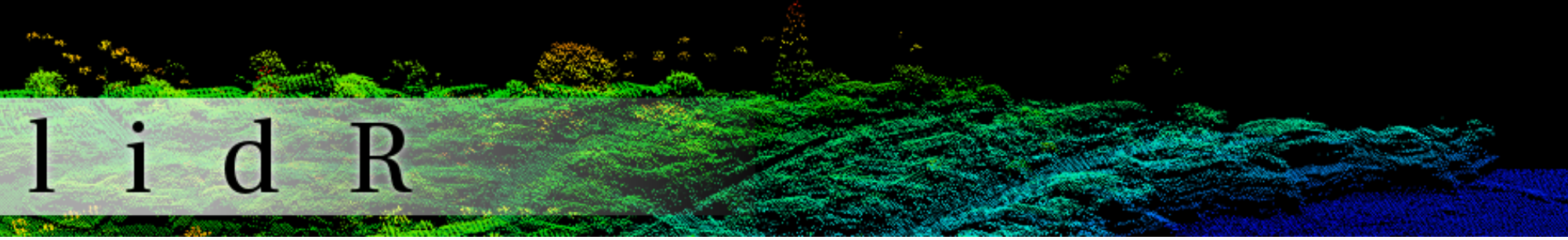
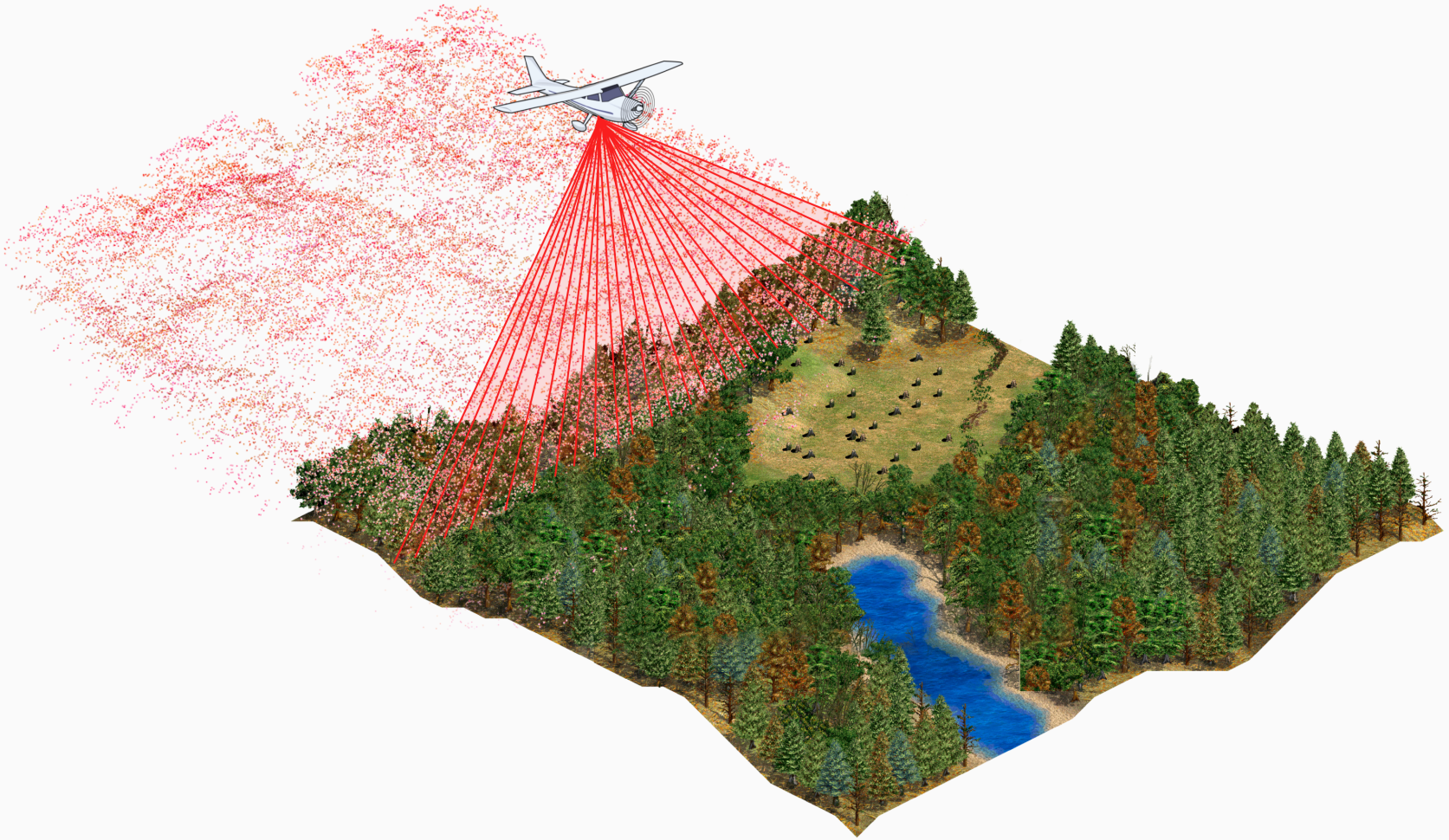


lidR, an R package to work with laser scanning point cloud in forestry and ecology

Roussel Jean-Romain, David Auty

LiDAR data

LiDAR is big data in many ways $\longrightarrow$ need advanced and efficient algorithms to analyse them.



Introduction – why an R package?

Why an R package?

1. An integrated framework: no switching environments

Why an R package?

1. An integrated framework: no switching environments
2. As researchers we must use open-source tools

Why an R package?

1. An integrated framework: no switching environments
2. As researchers we must use open-source tools
3. More flexible, more interactive, more tunable

Why an R package?

1. An integrated framework: no switching environments
2. As researchers we must use open-source tools
3. More flexible, more interactive, more tunable
4. Multi-platform: GNU/Linux, Mac OS, Windows

Why an R package?

1. An integrated framework: no switching environments
2. As researchers we must use open-source tools
3. More flexible, more interactive, more tunable
4. Multi-platform: GNU/Linux, Mac OS, Windows

Why an R package?

1. An integrated framework: no switching environments
2. As researchers we must use open-source tools
3. More flexible, more interactive, more tunable
4. Multi-platform: GNU/Linux, Mac OS, Windows

First presentation but...

...already used by several users in USA, Italy, France, Germany, Australia, Canada, Rwanda?, ...

Why an R package?

1. An integrated framework: no switching environments
2. As researchers we must use open-source tools
3. More flexible, more interactive, more tunable
4. Multi-platform: GNU/Linux, Mac OS, Windows

First presentation but...

...already used by several users in USA, Italy, France, Germany, Australia, Canada, Rwanda?, ...

ALS vs. TLS

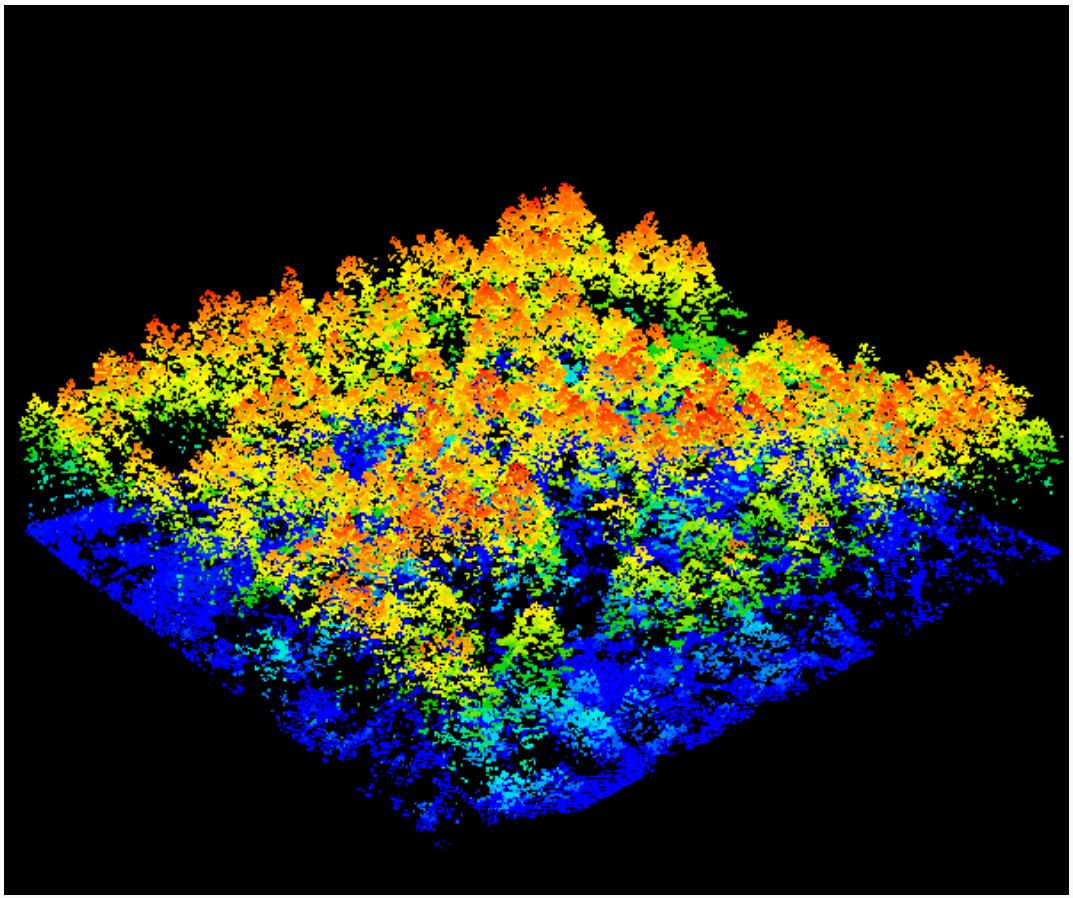
Designed for ALS but already used for TLS

Pre-processing tools

Read and write las and laz file in R

Supports of .las and .laz files in I/0

```
1  data = readLAS("myfile.laz")
2  plot(data)
```

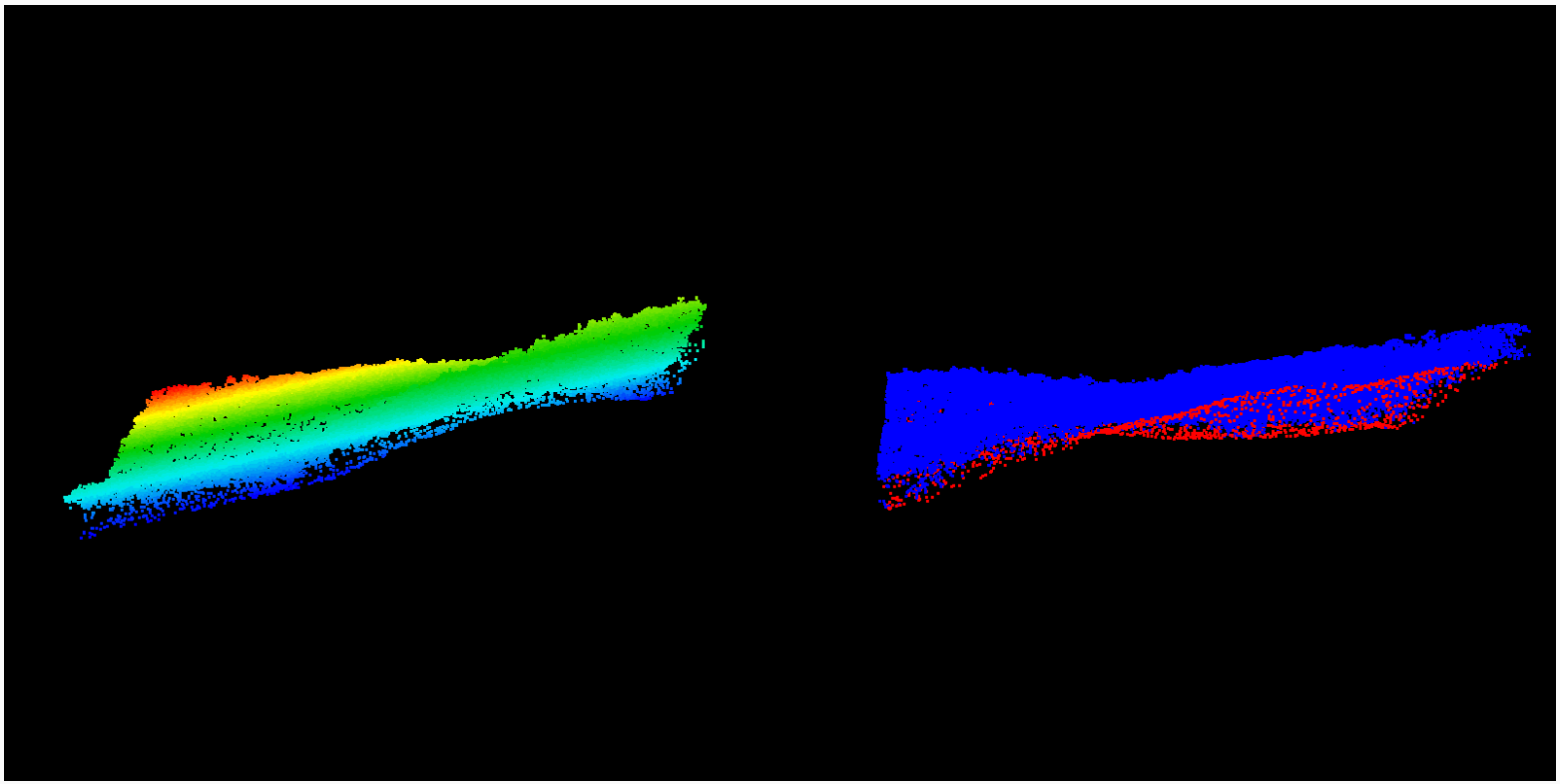


Ground classification

1 algorithm in v1.2.1

- Progressive morphological filter

```
1  lasground(data, ...)
2  plot(data, color = "Classification")
```

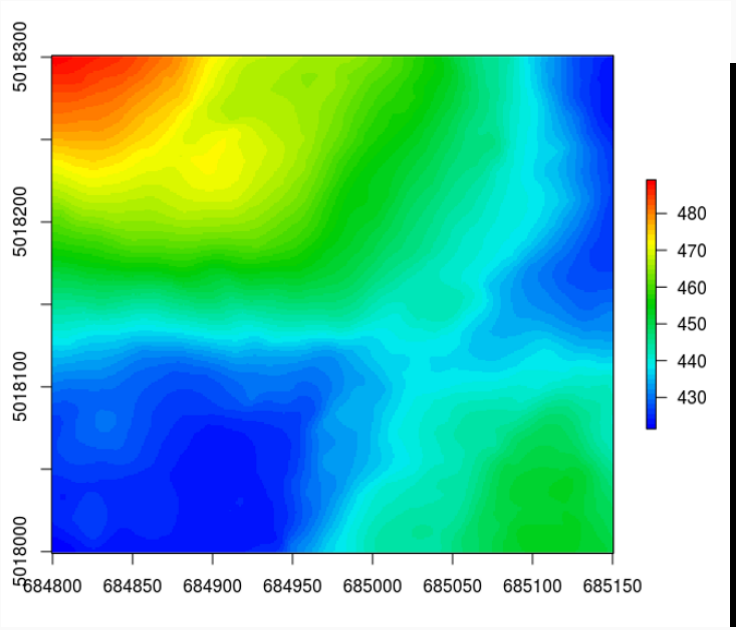


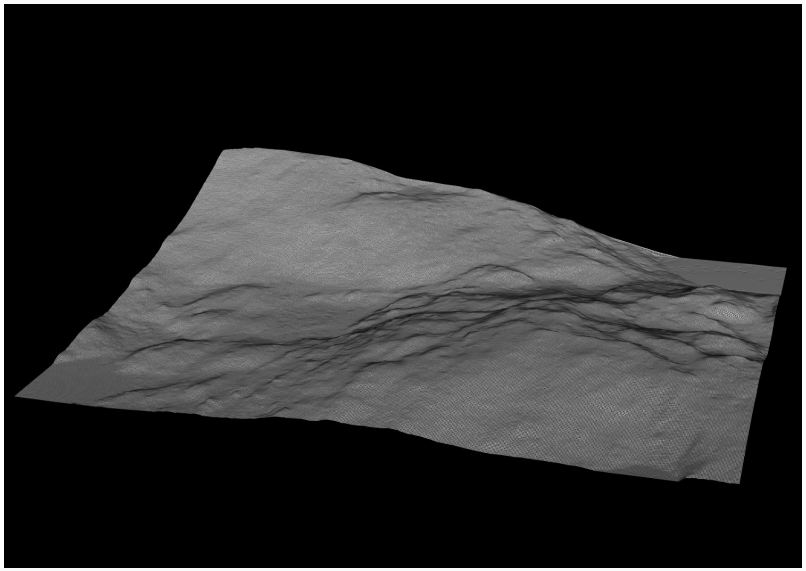
Terrain model

3 algorithms in v1.2.1 (expected 4 in v1.3.0)

- k-nearest neighbours + invert distance weighting
- Delaunay triangulation with linear interpolation
- k-nearest neighbours + kriging

```
1  dtm = grid_terrain(data, ...)
2  plot(dtm)
3  plot3d(dtm)
```



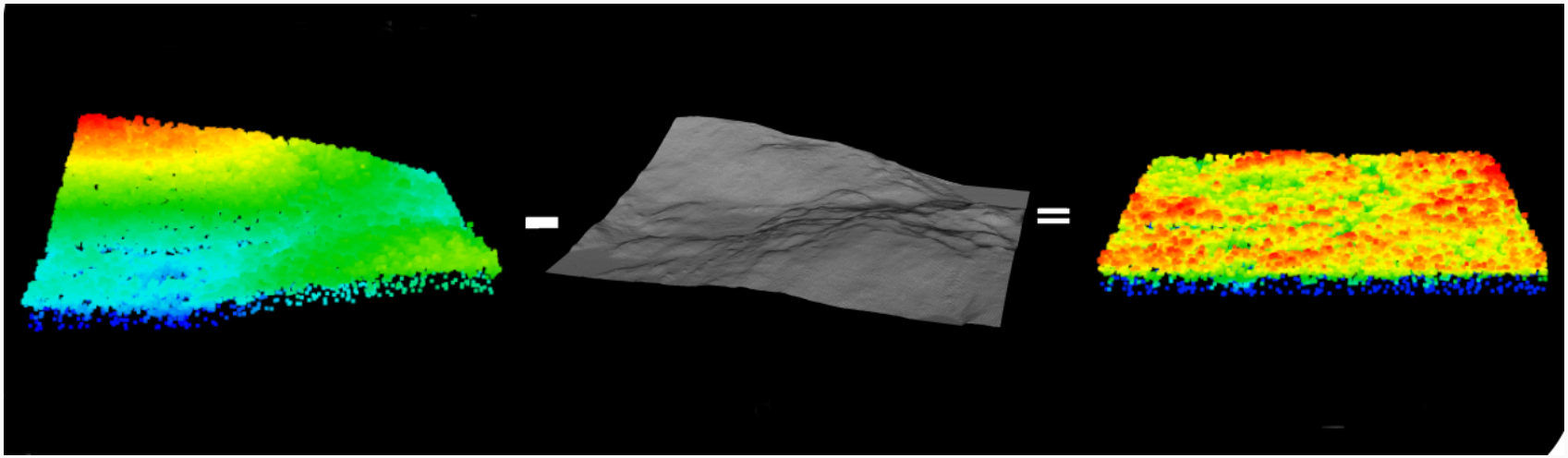


Normalization

2 algorithms in v1.2.1

- Discretized normalization: data - dtm
- True interpolation: interpolation of each point

```
data_norm = lasnormalize(data, ...)
plot(data_norm)
```

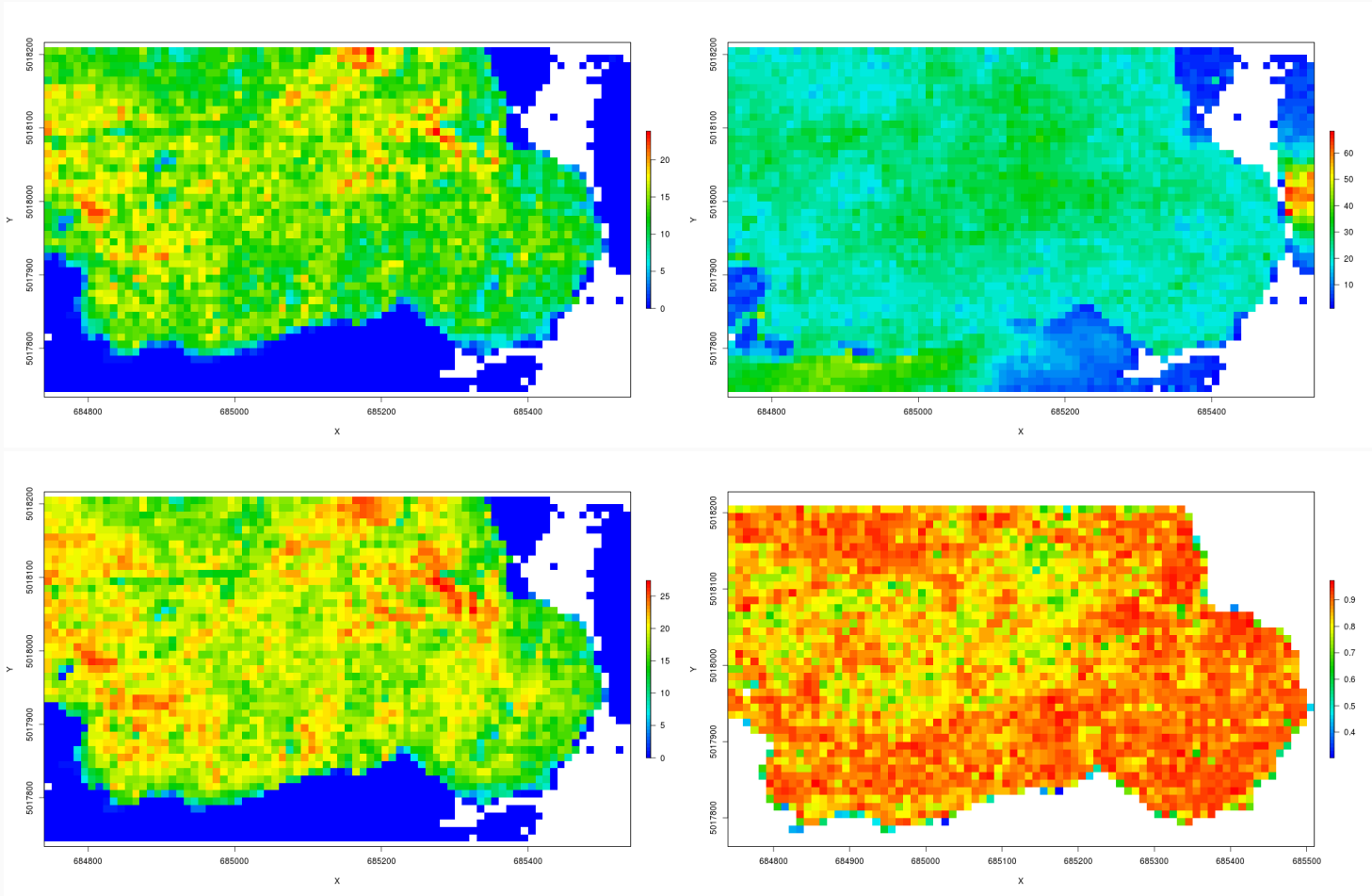


Area-based approach tools

Area-based approach (1)

A general assesment of the forest cover with a low to medium resolution.

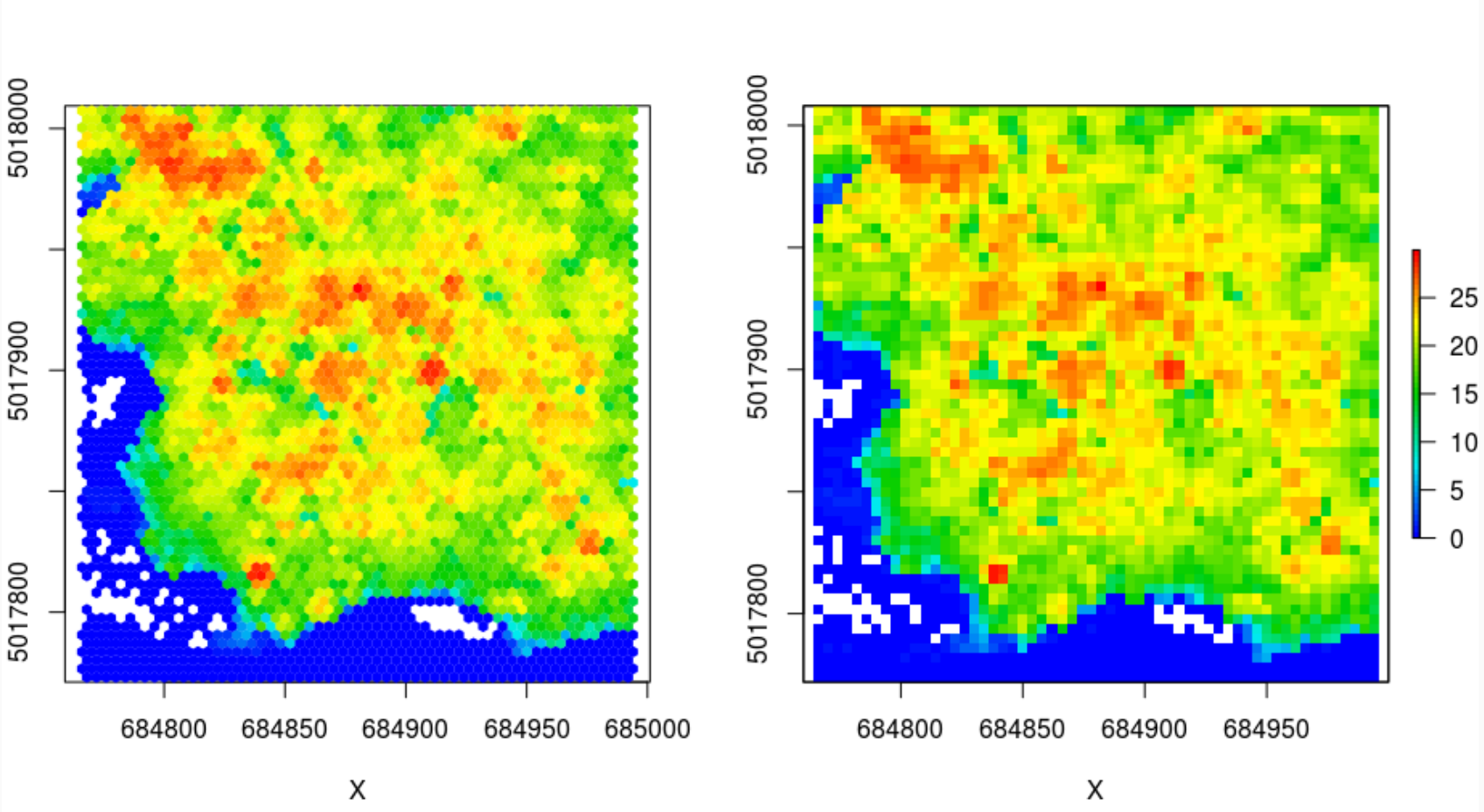
```
1  metrics = grid_metrics(data, ...)
2  plot(metrics)
```



Area-based approach (2)

Why do we use square pixels (rasters)?

```
1  metrics = grid_hexametrics(data, .sdtmetrics)
2  plot(metrics, "hmax")
```



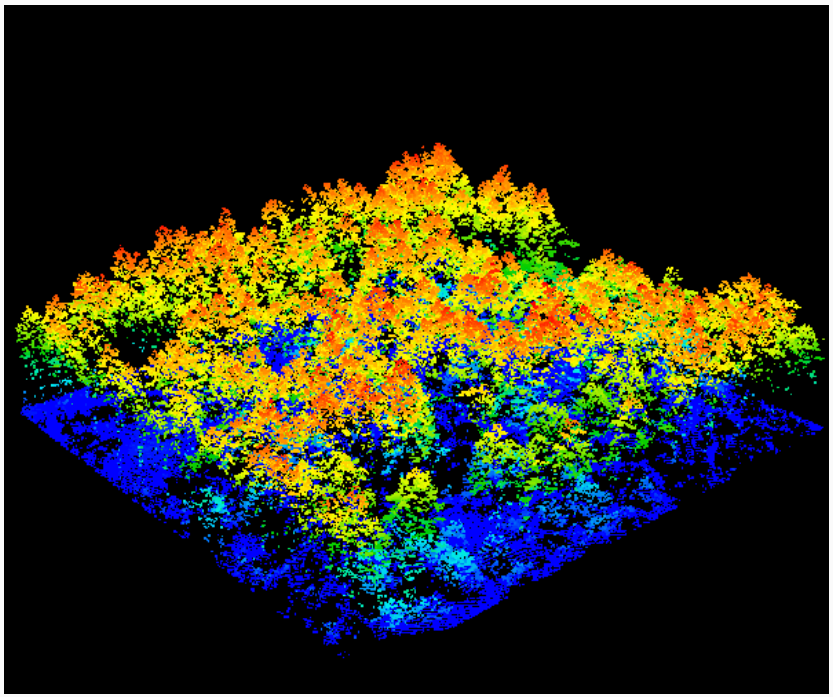
Canopy height models

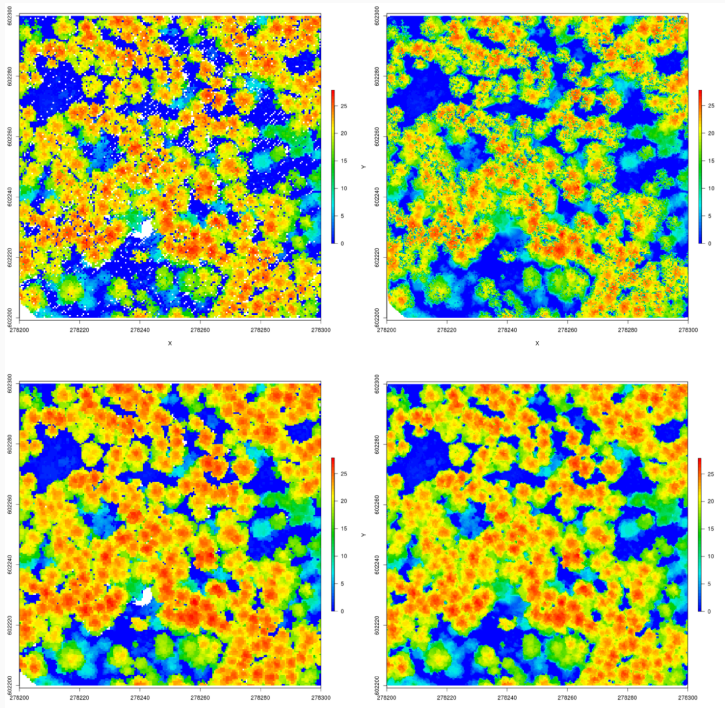
Canopy height models

3 algorithms (+ sugar!) in v1.2.1

- Grid local highest point (+ subcircling + hole filling)
- Triangulation of first returns (+ subcircling)
- Krosvavipour pit-free algorithm (+ subcircling)

```
1  chm = grid_canopy(data, ...)
2  plot(chm)
```





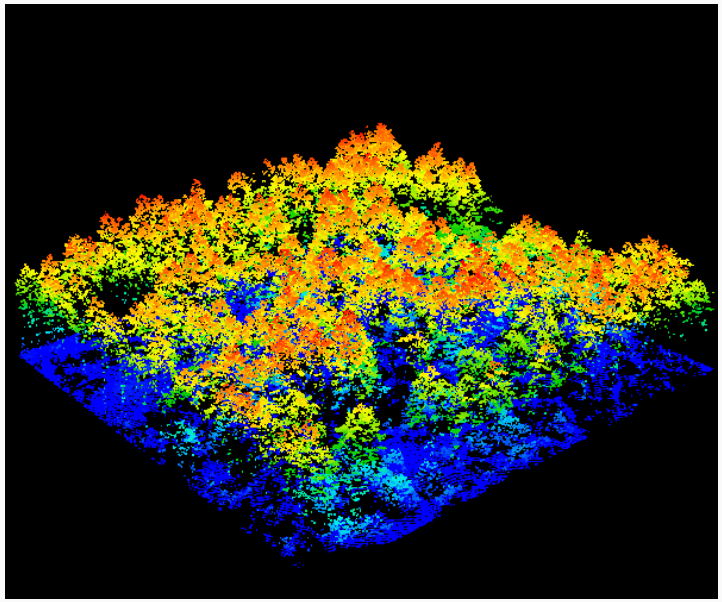
Individual tree segmentation tools

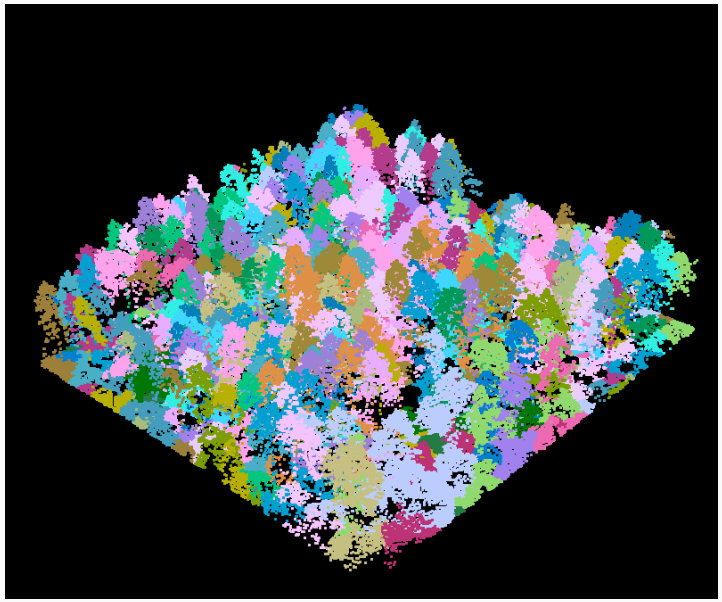
Individual tree segmentation (ITS)

3 algorithms in v1.2.1 (expected 4 in v1.3.0)

- Dalponte 2016 (canopy height model)
- Watershed (canopy height model)
- Li 2012 (point cloud level)

```
lastrees(data, ...)
plot(tree, color = "treeID", palette = pastel.colors(100))
```





Project management tools

Catalog tools

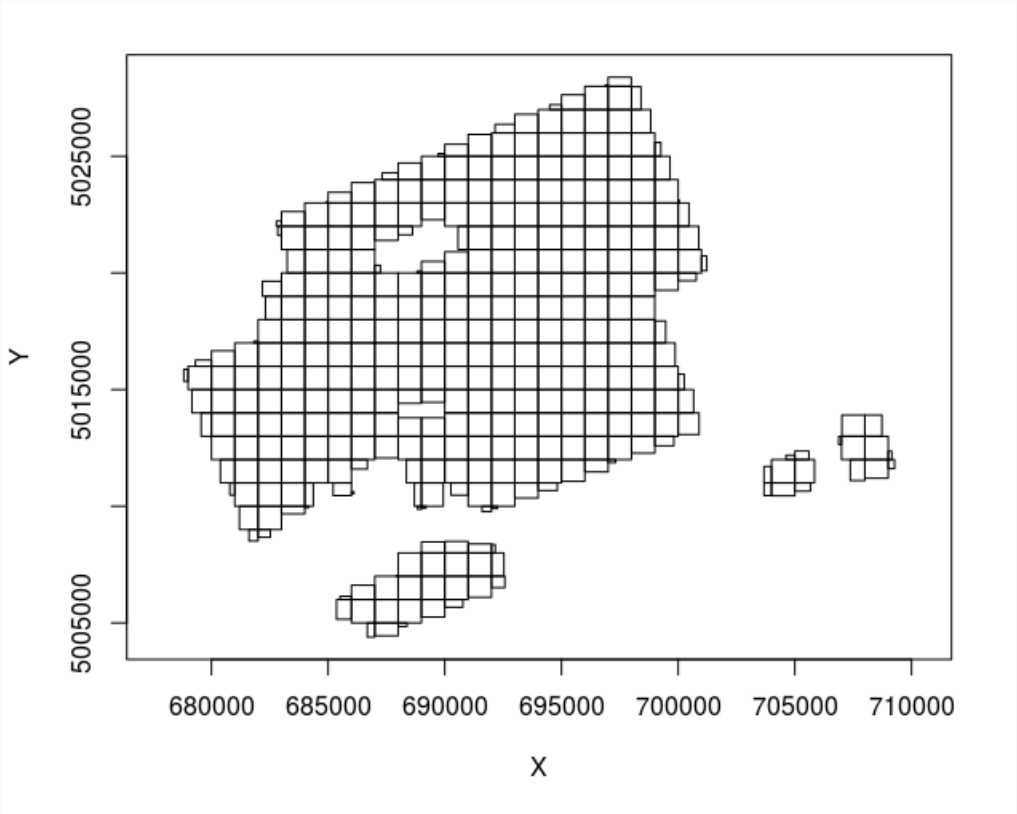
How to deal with an entire project? Use `catalog_*` tools.

```
ctg = catalog("<path to a folder>")
plot(ctg)
```

Catalog tools

How to deal with an entire project? Use `catalog_*` tools.

```
1  ctg = catalog("<path to a folder>")
2  plot(ctg)
```



Extract ground inventories

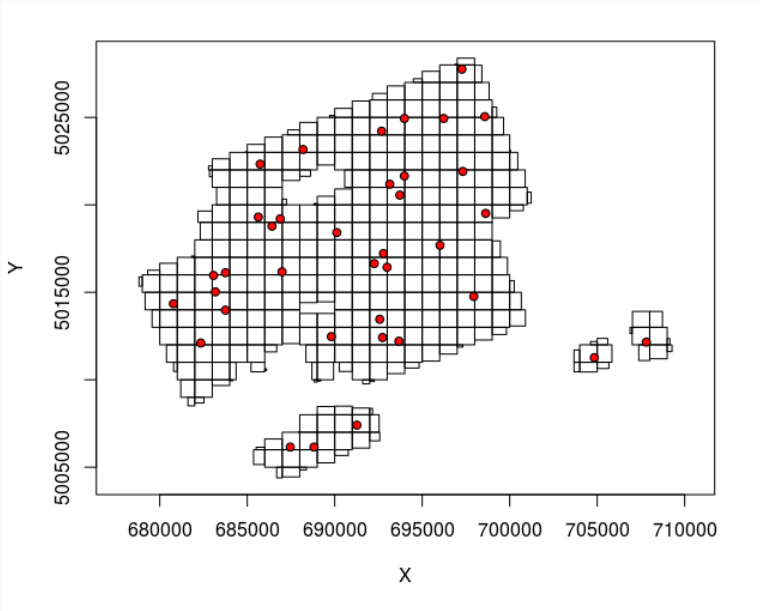
And many other uses...

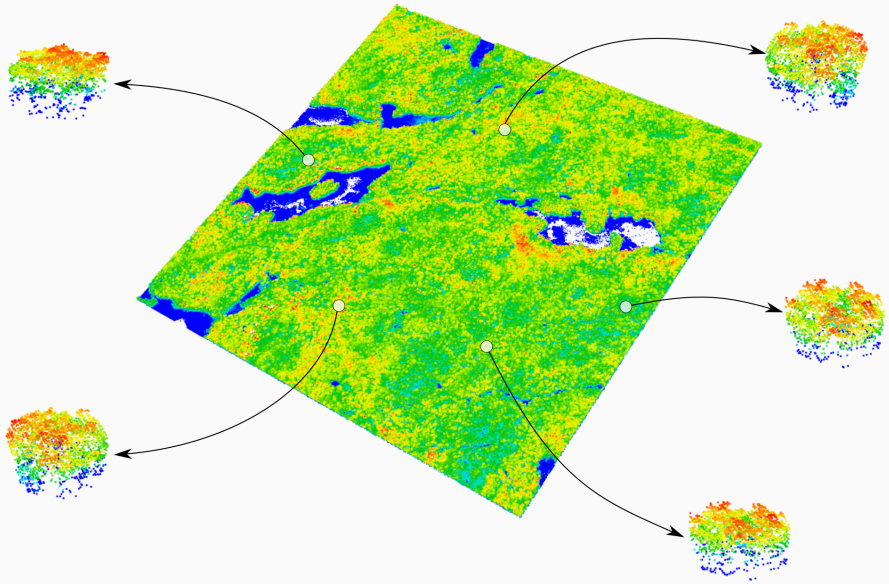
```
1  rois = catalog_queries(ctg, x, y, r)
```

Extract ground inventories

And many other uses...

```
1  rois = catalog_queries(ctg, x, y, r)
```





Other tools

Other tools

- Merge geographic data (raster, shapefile) at the point cloud level
- Retrieve individual flightlines
- Filters
- Colourize a point cloud
- Leaf area density, Rumple index (surface roughness), ...
- Grid point density
- Point decimation and homogenization
- Voxelization and metrics
- Streaming filters
- ...

`lidR` does not aim to provide predefined processes, it is designed to enable users to build their own.

Resources, bug reports, feature requests, colaborations

- https://github.com/Jean-Romain/lidR/
- https://github.com/Jean-Romain/lidR/wiki
- https://CRAN.R-project.org/package=lidR