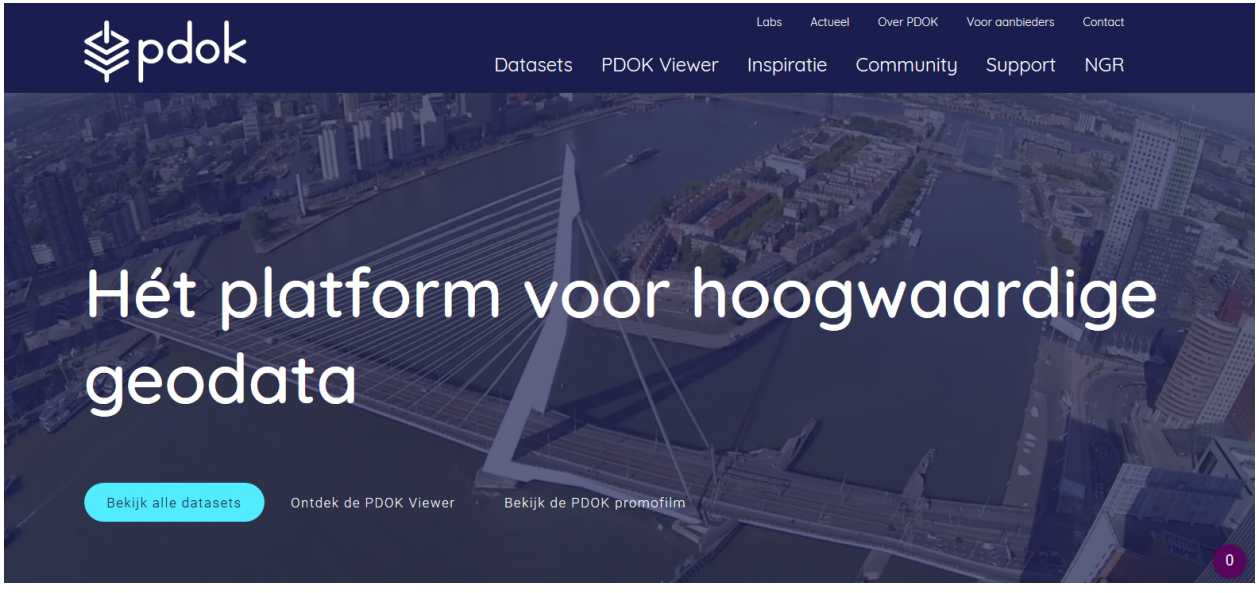
The nlgeocoder package

10 september 2018

Aim

Create an R interface to the PDOK Locatieserver API.
Locatieserver is a (reverse) geocoding service for The Netherlands.
Locatieserver is part of the Dutch National Spatial Data Infrastructure and hosted by PDOK.



Team

- Edwin de Jonge
- Egge-Jan Pollé
- Willy Tadema
- Juris Breidaks

Documentation

- https://github.com/PDOK/locatieserver
- https://forum.pdok.nl/c/applicaties-en-diensten/locatieserver

How do we name the package?

Suggestions:

- pdokr
- nllocation
- dutchgeocoder
- nlgeocoder

Considerations:

- The name should describe the functionality well
- Users should see immediately that the package is only for geocoding addresses in the Netherlands
- Maybe we want to create an interface to other PDOK functionality later, but for now we choose to create a compact package that only uses Locatieserver
- Prefix ‘nl’ is better than ‘dutch’, because the package will only work for The Netherlands, not for other countries where Dutch is spoken.
- The package should preferrably end with an ‘r’ ;-)

What license?

"As open as possible"

Naming conventions

- Prefix functions with `nl_`

Ideas

- Return the response as an `sf` object
- Use the `score` attribute as a thresshold value to determine if a search result should be returned to the user. If the score is too low, it is may not be a good fit.
- Mimic the `geocode` function in the `ggmap` package. For example, add a parameter `output` (`latlon`, `latlona`, `more`, `all`)
- Add a parameter to choose between returning the location in WGS84 or RD_New.
- Add a parameter to limit the number of rows returned with default value 1.
- Add a parameter to add the API key to the request. Apparantly not obligatory, but nice to have for PDOK to monitor API usage.
- Add parameter `type` with predefined values to choose from (e.g. `adres`).

Findings

- You can only do one-by-one requests to the Locatieserver, no bulk requests.
- What are the terms of use for Locatieserver? There are none. See this answer in the PDOK forum.

Tasks

- Create unit tests with `testthat` (Willy)
- Add documentation using Roxygen2
- Add functionality and make it more user-friendly (Edwin)
- Create demo using tmap and shiny (Egge-Jan)