R / Python and Big Data; openEO

Edzer Pebesma

EDC FORUM 2017
Big Data Analytics & GIS  September 21-22, Münster
Who am I?

- User, and contributor, of open source software since 1997
- Active member of the R community since 2003
- Professor at ifgi since 2007
- Editor for the Journal of Statistical Software, and Computers & Geosciences
- actively in search for the meaning of open science
Open Science (or: why I don’t use ESRI software)

- transparency is a key pillar in science: everything needs to be questioned, all details need to be scrutinized
- in geoinformatics, an important component of research is the computational manipulations of data (“from data to information”)
- only open source software fully discloses all details of scientific computation
- equivalent to how open access opens access to published text, open source software reveals the details of computational procedures underlying scientific findings
- ESRI’s take on “open” (search: “ESRI’s open vision”):
  - commit to interoperability
  - let users share open data and collaborate
  - promote open source software that binds to ESRI’s software
- these goals are fine to engineer solutions, but not sufficient for open science
Reproducibility

- Reproducibility is an important aspect of scientific research, because the credibility of science is at stake when research is not reproducible (Pebesma/Nüst/Bivand 2012, Eos 93(16), 163-164).
- DFG guidelines of good scientific practice require researchers to be able to reproduce all findings at least 10 years after finishing each funded project.
- In practice this means that along with a paper, we have to share and archive:
  - the data used
  - the code or scripts used
  - the runtime (OS, executables)
- This is not a problem, unless you use licensed software and proprietary OSs.
What can we do?

- use scripting languages; for data science: R, Python, Julia
- use literate programming (R Sweave / R markdown / Jupyter notebooks, etc.)
- reuse (open source) software that others use
- develop, and share, software that others benefit from
- publish methods, but also about software!

⇒ this all contributes to a shared understanding of science, and by that of our world
What can we do?

- use scripting languages; for data science: R, Python, Julia
- use literate programming (R Sweave / R markdown / Jupyter notebooks, etc.)
- reuse (open source) software that others use
- develop, and share, software that others benefit from
- publish methods, but also about software!

⇒ this all contributes to a shared understanding of science, and by that of our world
Big data: what is it?

It is: ... big. Meaning: large in volume.
- it doesn’t fit on your computer
- it doesn’t fit on your large computer
- it takes long to summarize to small data
- it’s hard to interact with
- most data you and I work with is not big.

Of the existing non-spatial solutions, what are they used for?
Big data: what is it?

It is: ... big. Meaning: large in volume.
▶ it doesn’t fit on your computer
▶ it doesn’t fit on your large computer
▶ it takes long to summarize to small data
▶ it’s hard to interact with
▶ most data you and I work with is not big.

Of the existing non-spatial solutions, what are they used for?
Good CS expert says: Most firms that thinks they want advanced AI/ML really just need linear regression on cleaned-up data.

7:19 PM - 28 Nov 2016

1,038 Retweets 1,739 Likes

Anders Sandberg @anderssandberg · 28 Nov 2016
Replied to @robinhanson
A hedge fund tried to hire me on the spot when I suggested a regression instead of using IBM Watson on their dataset.

Anders Sandberg @anderssandberg · 28 Nov 2016
Maybe it was secret test: hire people who suggest something sensible instead of latest fashion. But given signalling I doubt it

robin hanson @robinhanson · 28 Nov 2016
People who say something sensible are probably so rare that it isn't worth designing special tests to find them.
Spatial / spatiotemporal data

- most programming languages have built-in support for time data, but not for spatial data
- most statistical and ML methods assume independent observations: order of records in a table doesn’t matter; many spatiotemporal methods cannot assume this
- is the algorithm embarrassingly parallel? CA: chunking-averaging (Matloff 2016, JSS 74(4)); map-reduce.
- non-trivial: image segmentation, watershed deliniation, routing through network, interacting agents (Th. Paschke)
- how to chunk: by pixel (time series), by time (scene collection), or both?
Python

- pandas: Python Data Analysis Library (DataFrame, time series)
- Dask: a flexible parallel computing library for analytic computing.
- GeoPySpark: a Python language binding library of the Scala library GeoTrellis.
- xarray: N-D labeled arrays and datasets (CDM, NetCDF; integrates well with Dask)
- general purpose; distributed community;
- from osgeo import gdal
Julia

- belongs in the data science scripting languages tripple “R - python - Julia”
- quite young
- learned from experiences in R and python
- emphasis on performance
- relatively little usage; research stage?
R

- R is a free software environment for statistical computing and graphics.
- oldest of the three; originally written for interactive analysis (REPL), rather than performance
- data in memory; upcoming ALTREP changes this
- R is extendible; on CRAN, over 11,000 extension packages
- most developers are users too
- CRAN task views of interest: HighPerformanceComputing, Spatial, SpatioTemporal, TimeSeries
- strong, friendly and centered community
R for big data

- **dplyr**: interface to tabular data, internal, or external: PostgreSQL, MariaDB, MonetDB, Impala, Spark, Hyve, BigQuery, ... ; translates R expressions to SQL
- **matter**: (BioConductor): a framework for rapid prototyping with binary data on disk
- **parallel**: multicore, multi-workstation (incl. MPI)
- (direct spark or hadoop interfaces)

... for big spatiotemporal data:
- **dplyr**: might work for spatial databases
- **SciDBR, scidb4geo**: interface to SciDB array database
- **raster**: data cube (3D) from (band/z/t) collection of tiles; on-disk
- **stars**: R Consortium funded project (under development; e.t.a. 2018); includes remote storage and computing
- **openEO.org**
> x
[1] "avhrr-only-v2.19810901.nc"  "avhrr-only-v2.19810902.nc"
[3] "avhrr-only-v2.19810903.nc"  "avhrr-only-v2.19810904.nc"
[5] "avhrr-only-v2.19810905.nc"  "avhrr-only-v2.19810906.nc"
[7] "avhrr-only-v2.19810907.nc"  "avhrr-only-v2.19810908.nc"
[9] "avhrr-only-v2.19810909.nc"
> (y = st_stars(x, quiet = TRUE))
stars object with 4 dimensions and 4 attributes
attribute(s):
Min. : -1.8  Min. : -8.2  Min. : 0.1  Min. : 0
1st Qu.: 1.4  1st Qu.: -0.5  1st Qu.: 0.1  1st Qu.: 1
Median :14.3  Median : 0.0  Median : 0.3  Median : 1
Mean :13.7  Mean : -0.1  Mean : 0.3  Mean : 1
3rd Qu.:25.1  3rd Qu.: 0.4  3rd Qu.: 0.3  3rd Qu.: 1
Max. :33.9  Max. : 5.6  Max. : 1.0  Max. : 1
NA’s :3110850  NA’s :3110850  NA’s :3110850  NA’s :8094523
dimension(s):
  from to offset delta refsys
x 1 1440 0 0.25 +proj=longlat +datum=WGS84 +no_defs
y 1 720 90 -0.25 +proj=longlat +datum=WGS84 +no_defs
time 1 9 1981-09-01 1 days POSIXct
zlev 1 1 0 meters NA
ArcGIS-R bridge

- R is (mostly) GPL, how can closed source software link to it?
- (ESRI’s) lawyers say it’s a gray zone, but think it is all right (...)
- Why doesn’t ArcGIS bridge to TERR (Tibco Enterprise R Runtime)? Customers don’t like to pay for that.
- If you contribute to R-ArcGIS bridge functions, you contribute to ESRI, and not to Open Science
openEO.org

A Common, Open Source Interface between Earth Observation Data Infrastructures and Front-End Applications

- H2020 project funded under call EO-2-2017: EO Big Data Shift
- TU Wien (Coordinates), ifgi, WUR, VITO, EODC, Mundialis, Sinergise, EURAC Research, Solenix, JRC, (Google)
- background: EO data are too large to download
- we all work on the same satellite imagery, but how do R/python/javascript users access these?
- heterogeneity: choosing one cloud platform is such an investment that nobody validates outcomes against another
- choosing a set of abstractions (data models) and interfaces (processes), implement use cases against different backends
- side effect: makes cloud offerings comparable, in terms of data, functionality and costs

http://r-spatial.org/2016/11/29/openeo.html