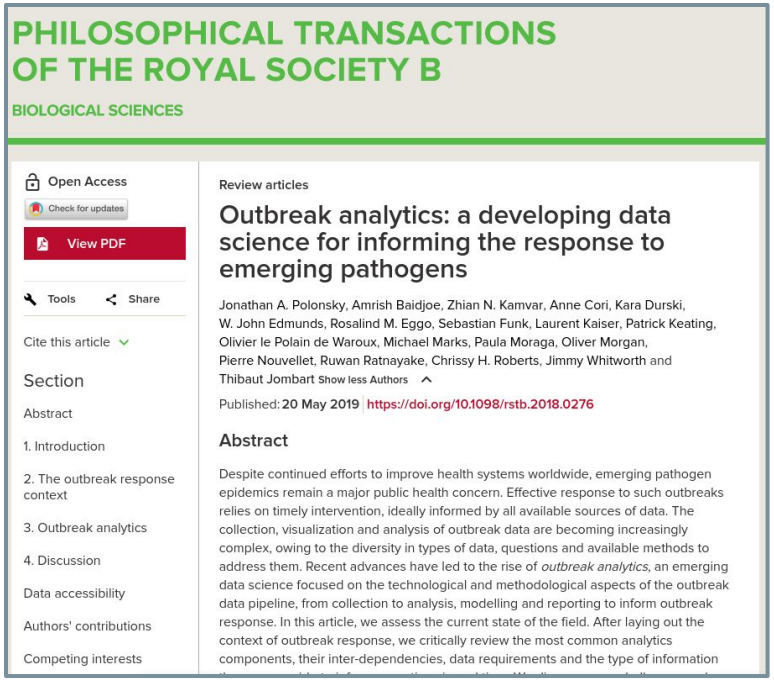
Tools for outbreak analytics infrastructures

Thibaut Jombart (@TeebzR)
London School of Hygiene and Tropical Medicine
Imperial College London
R Epidemics Consortium (RECON)

Last update:
14 November 2019

Context

On the emergence of “outbreak analytics”



PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY B

BIOLOGICAL SCIENCES

Open Access

Check for updates

View PDF

Tools Share

Cite this article

Section

Abstract

1. Introduction

2. The outbreak response context

3. Outbreak analytics

4. Discussion

Data accessibility

Authors' contributions

Competing interests

Review articles

Outbreak analytics: a developing data science for informing the response to emerging pathogens

Jonathan A. Polonsky, Amrish Baidjoe, Zhian N. Kamvar, Anne Cori, Kara Durski, W. John Edmunds, Rosalind M. Eggo, Sebastian Funk, Laurent Kaiser, Patrick Keating, Olivier le Polain de Waroux, Michael Marks, Paula Moraga, Oliver Morgan, Pierre Nouvellet, Ruwan Ratnayake, Chrissy H. Roberts, Jimmy Whitworth and Thibaut Jombart Show less Authors

Published: 20 May 2019 https://doi.org/10.1098/rstb.2018.0276

Abstract

Despite continued efforts to improve health systems worldwide, emerging pathogen epidemics remain a major public health concern. Effective response to such outbreaks relies on timely intervention, ideally informed by all available sources of data. The collection, visualization and analysis of outbreak data are becoming increasingly complex, owing to the diversity in types of data, questions and available methods to address them. Recent advances have led to the rise of *outbreak analytics*, an emerging data science focused on the technological and methodological aspects of the outbreak data pipeline, from collection to analysis, modelling and reporting to inform outbreak response. In this article, we assess the current state of the field. After laying out the context of outbreak response, we critically review the most common analytics components, their inter-dependencies, data requirements and the type of information

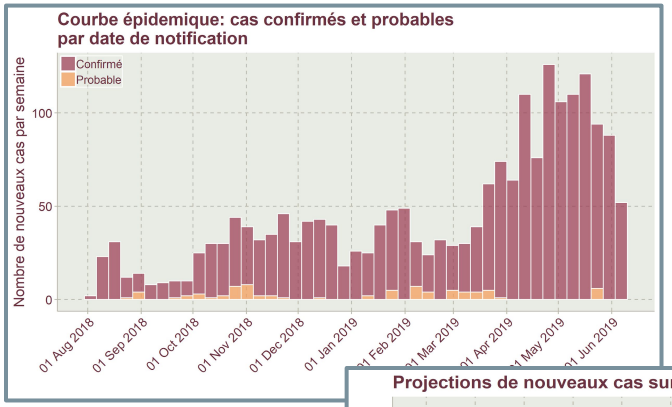
https://doi.org/10.1098/rstb.2018.0276

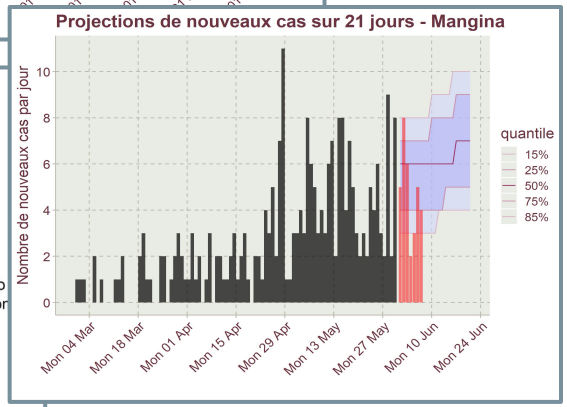
- DoB: Polonsky et al. (2019) Phil. Trans. R. Soc. B 374
- Data science mixing statistics, mathematical modeling, computer simulations, database infrastructure, GIS, genetics, software engineering
- At the crossroad of public health institutions, private sector, and academia
- Aims to inform response to emergencies in real-time

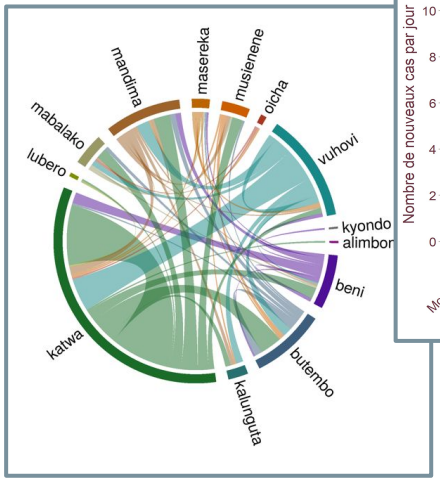
Ebola in North-Kivu & Ituri, DRC

- Largest Ebola epidemic in DRC, 2nd largest in the world
- August 2018 - 14 Nov 2019:
  - >3,200 cases (confirmed / probable)
  - 67% deaths
- Difficulties due to insecurity and armed conflicts
  - Threats to local population
  - Threats to response staff and facilities
- **First deployment of an analytical cell** as part of the Emergency Operations Centre

Outbreak analysis cell: aims and challenges







- Multiple (messy) data sources, no global database
- Independent updates of different databases
- Needs: data cleaning, visualisation, in-depth analyses, forecasting
- Routine versus *ad-hoc* analyses
- Need for regular results updates and traceability
- Bad internet, different platforms, low R literacy

Tools for outbreak analytics infrastructures

Tidier markdown workflows with *reportfactory*
Data cleaning using *linelist*
Taking R offline: the RECON *deployer*

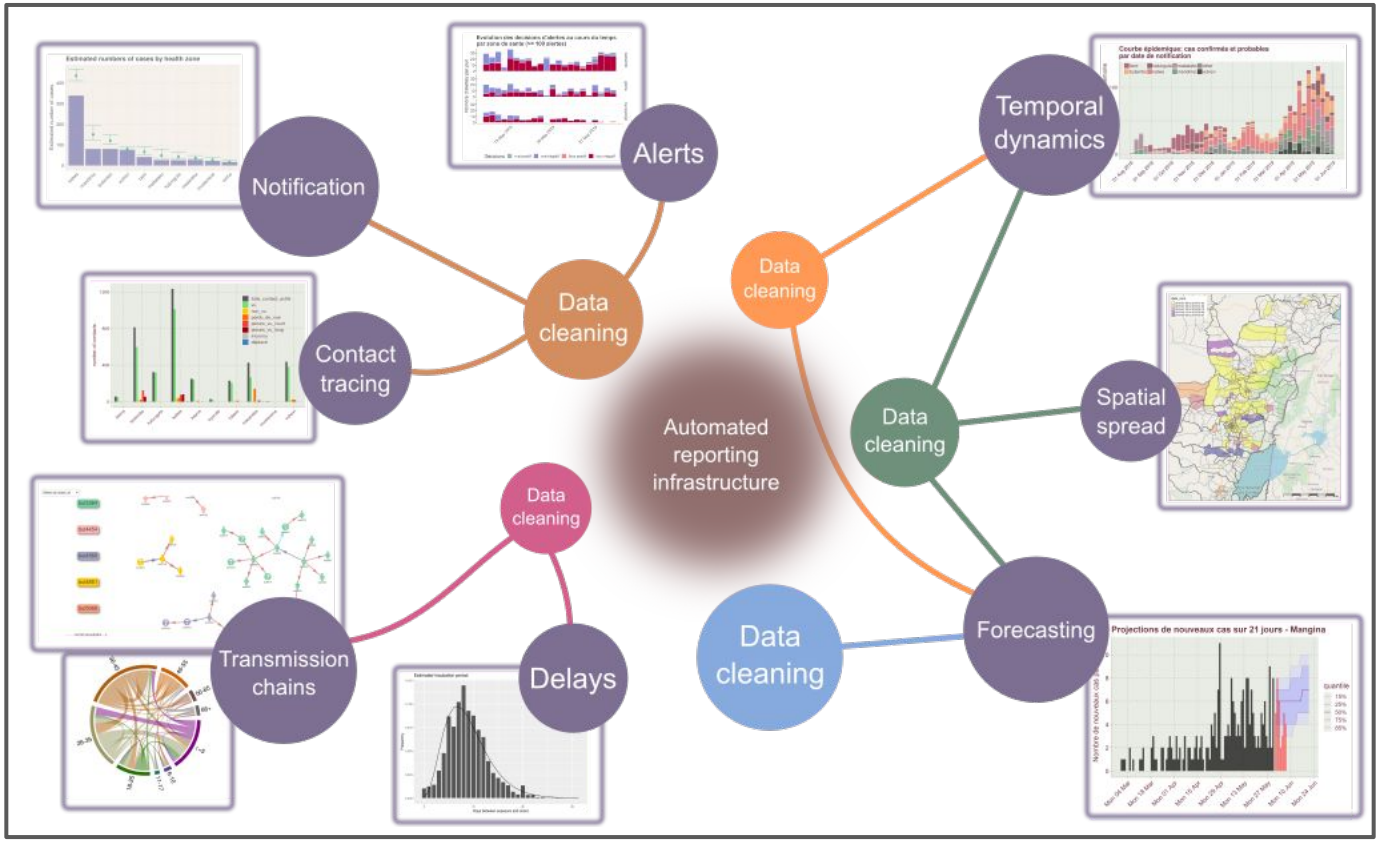
Tools for outbreak analytics infrastructures

Tidier markdown workflows with *reportfactory*

Data cleaning using *linelist*

Taking R offline: the RECON *deployer*

Example: analysis infrastructure of Ebola response, DRC, 2019

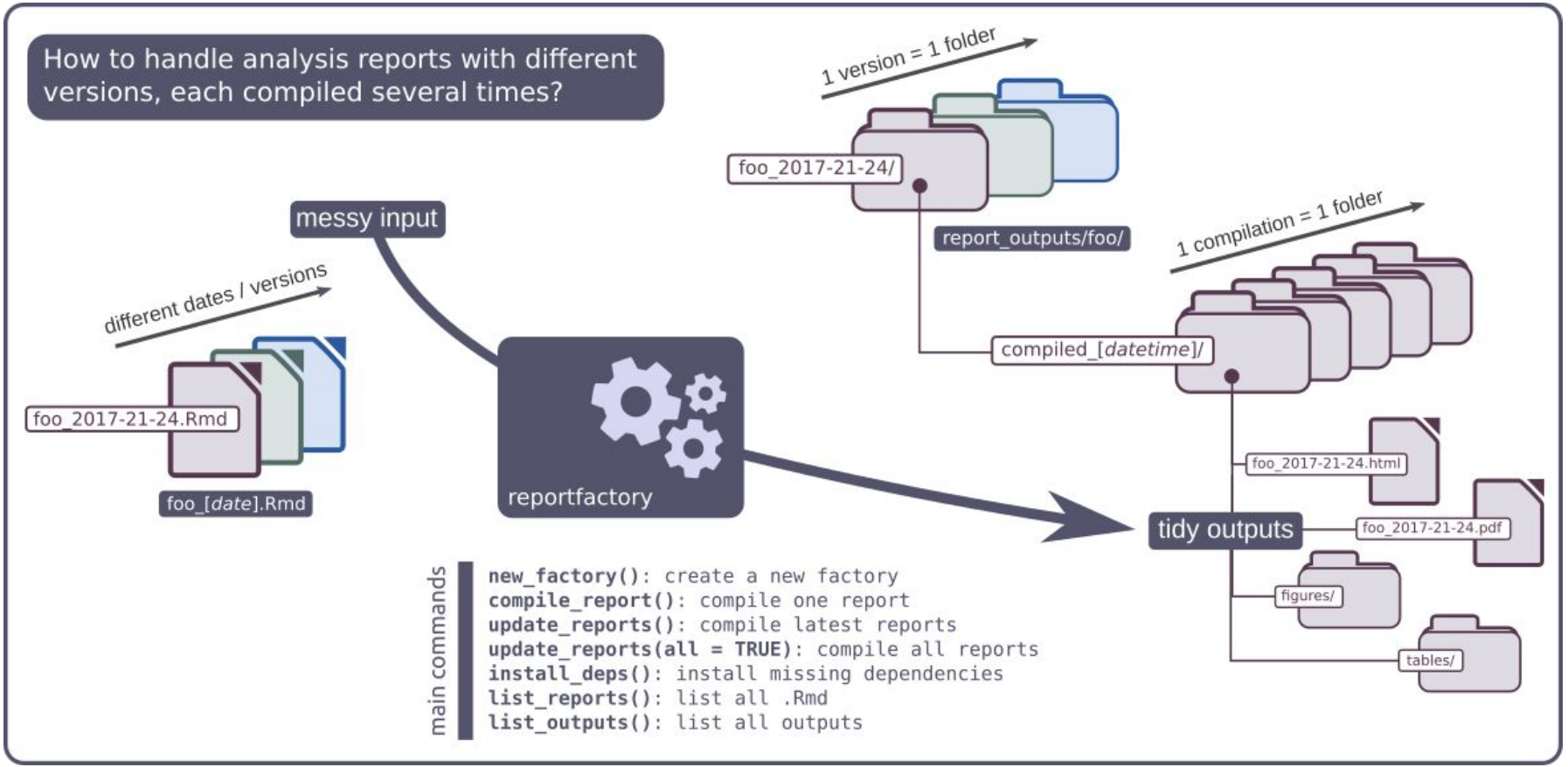


Tidier rmarkdown workflows with *reportfactory*: use case

Original requirements

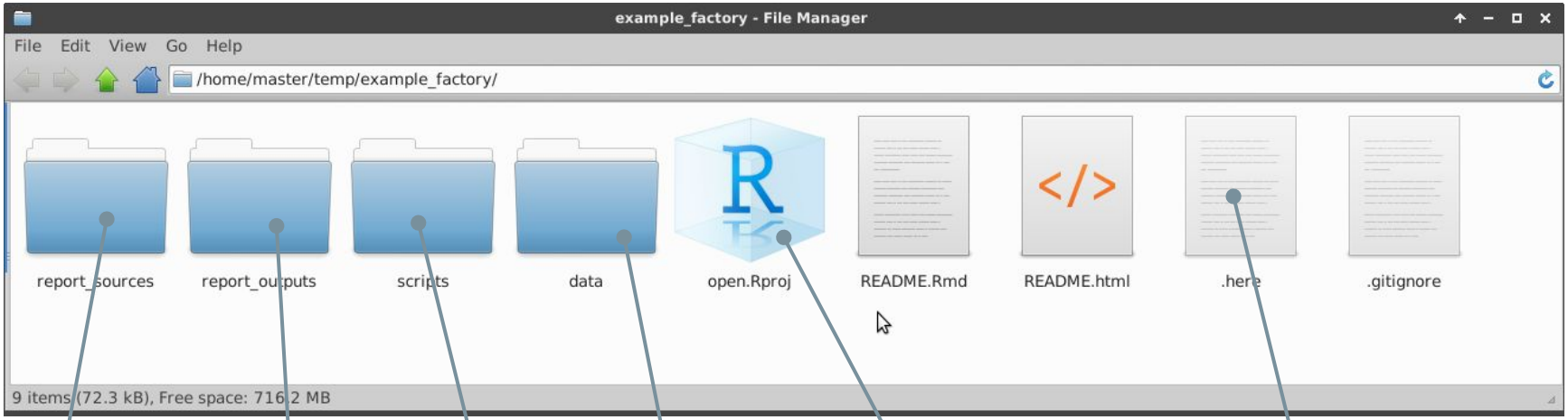
- Handle multiple `.Rmd` reports
- Handle multiple (dated) versions of the same report
- Separate data, scripts, `.Rmd` sources, outputs
- Generates time-stamped outputs
- Update all reports in one go
- Handle dependencies on packages
- Non-invasive: use of standard `.Rmd`, no config file
- Easy to use: accessible by people new to R
- Offline: does not require internet
- Portable: work on any platform

What does the *reportfactory* do?



reportfactory : basic structure

Creating a new factory: `new_factory()`



`.Rmd` files

Outputs: html files, figures etc

`.R` scripts

Data

Open the factory

Anchor (for file paths)

reportfactory : workflow

Once the factory is created:

1. Create dated report (e.g. `foo_2019-11-14.Rmd`)

   in `report_sources/`

2. Test output regularly using:

   ```
   rmarkdown::render("foo_2019-11-14.Rmd")
   ```

3. When happy with final version, open factory (`open.Rproj`) and type:

   ```
   library(reportfactory)
   compile_report("foo_2019-11-14.Rmd", clean_report_sources = TRUE)
   ```

4. Check outputs in `report_outputs/`

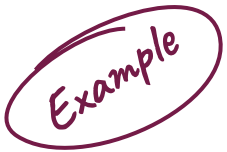
reportfactory : main functionalities

Other functionalities

- List / install dependencies: `list_deps() / install_deps()`
- List reports: `list_reports()`
- Compile all recent reports: `update_reports()`
- Compile specific report: `compile_report()`
- Archive old reports: `archive_reports()`
- … (suggestions welcome!)

reportfactory tricks #1: global scripts

- Global scripts are `.R` script files common to all reports in a factory, e.g. for loading required packages
- They are stored in `` `scripts/` `` or `` `src/` `` at the root of the factory
- They can be loaded inside an report using `rfh_load_scripts()`



In script file `scripts/aaa_load_packages.R`:

```
library(tidyverse)
library(linelist)
library(epicontacts)
```

In the *rmarkdown* source file `report_sources/foo_2019-11-14.Rmd`:

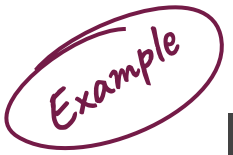
````
```{r load_scripts}

## this loads all scripts in /scripts/ including packages
## needed for the analysis
reportfactory::rfh_load_scripts()

```
````

reportfactory tricks #2 : parameterized reports

- Global variables can be passed to reports through `"params"` in `update_reports()` or `compile_report()`
- Can be used e.g. to generate separate reports for subsets of data



In the *rmarkdown* source file `report_sources/foo_2019-11-14.Rmd`:

````
```{r filter_data}

## filter linelist data by location if specified
if (exists("params") && !is.null(params$locations)) {
  linelist <- linelist %>%
    filter(health_zone %in% params$locations)
}

```
````

To run all analyses keeping only the health zones of `ankh` and `morpork`:

```
update_reports(params = list(locations = c("ankh", "morpork")))
```

Tools for outbreak analytics infrastructures

Tidier markdown workflows with *reportfactory*

Data cleaning using *linelist*

Taking R offline: the RECON *deployer*

Data standardisation using *linelist*

```
x %>% clean_data()
```

- Capitalisation
- Accents
- Separators
- Dates

'ID	Date of Onset.	GENDER_	Épi.Case_définition	messy/dates
khdntz	2018-01-09	male	Confirmed	that's 24/12/1989!
hmckhn	2018-01-09	male	suspected	// 24//12//1989
ekjmyd	2018-01-09	Female	confirmed	that's 24/12/1989!
kmoczh	2018-01-04	MALE	suspected	female
kftifx	2018-01-02	FEMALE	suspected	// 24//12//1989
qyipse	2018-01-09	Male	PROBABLE	01-12-2001
zprzec	2018-01-03	male	suspected	*NA*
bgsmfn	2018-01-06	Female	suspected	that's 24/12/1989!
syfnfd	2018-01-05	Female	confirmed	01-12-2001
aekdlv	2018-01-07	FEMALE	not a case	female
kcejly	2018-01-05	Female	Confirmed	that's 24/12/1989!
jyxnhl	2018-01-11	female	confirmed	// 24//12//1989

id	date_of_onset	gender	epi_case_definition	messy_dates
khdntz	2018-01-09	male	confirmed	1989-12-24
hmckhn	2018-01-09	male	suspected	1989-12-24
ekjmyd	2018-01-09	female	confirmed	1989-12-24
kmoczh	2018-01-04	male	suspected	*NA*
kftifx	2018-01-02	female	suspected	1989-12-24
qyipse	2018-01-09	male	probable	2001-12-01
zprzec	2018-01-03	male	suspected	*NA*
bgsmfn	2018-01-06	female	suspected	1989-12-24
syfnfd	2018-01-05	female	confirmed	2001-12-01
aekdlv	2018-01-07	female	not_a_case	*NA*
kcejly	2018-01-05	female	confirmed	1989-12-24
jyxnhl	2018-01-11	female	confirmed	1989-12-24

Dictionary-based cleaning using *linelist*

```
x %>% clean_data(wordlists = rules)
```

Typos
Re-levelling
Variable-specific rules

'ID	Date of Onset.	GENDER_	Épi.Case_définition
hlywxf	2018-01-10	m	ConFRImed
zgsjfx	2018-01-05	man	*NA*
nbmrvn	2018-01-08	female	*NA*
fasshf	2018-01-02	male	suspected
wlfhgk	2018-01-03	f	Not.a.Case
qdmhyp	2018-01-08	*NA*	Confirmed
ywntgm	2018-01-03	male	not a case
vlpamu	2018-01-04	male	PROBABLE
fqigws	2018-01-02	MALE	Not.a.Case
vrzpkj	2018-01-06	Female	confirmed
gsbjak	2018-01-06	f	female
zozxjp	2018-01-11	f	male

rules

change	to	variable
m	male	gender
f	female	gender
man	male	gender
.missing	unknown	.global
confrimed	confirmed	epi_case_definition
female	unknown	epi_case_definition
male	unknown	epi_case_definition

id	date_of_onset	gender	epi_case_definition
hlywxf	2018-01-10	male	confirmed
zgsjfx	2018-01-05	male	unknown
nbmrvn	2018-01-08	female	unknown
fasshf	2018-01-02	male	suspected
wlfhgk	2018-01-03	female	not_a_case
qdmhyp	2018-01-08	unknown	confirmed
ywntgm	2018-01-03	male	not_a_case
vlpamu	2018-01-04	male	probable
fqigws	2018-01-02	male	not_a_case
vrzpkj	2018-01-06	female	confirmed
gsbjak	2018-01-06	female	unknown
zozxjp	2018-01-11	female	unknown

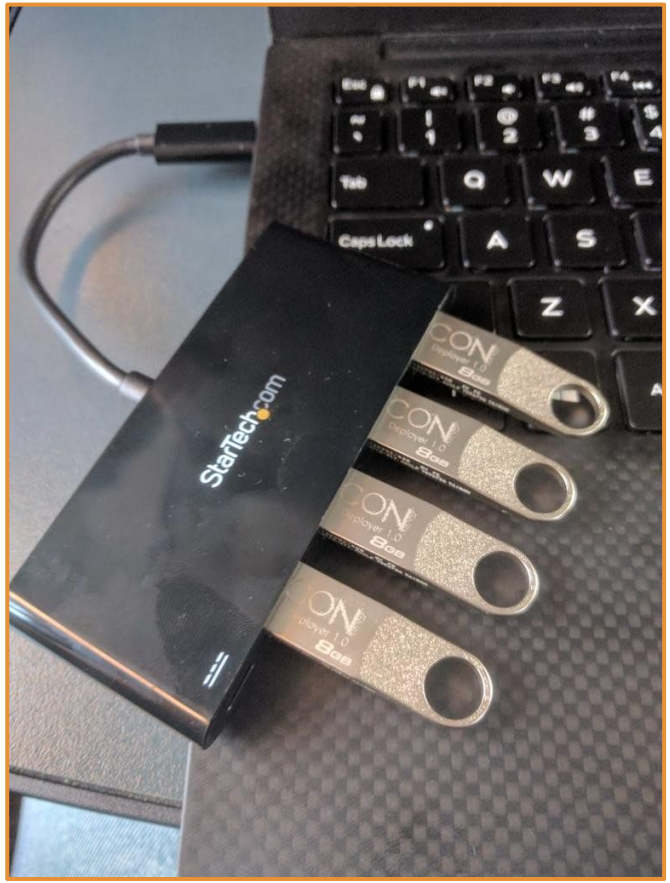
Tools for outbreak analytics infrastructures

Tidier markdown workflows with *reportfactory*

Data cleaning using *linelist*

Taking R offline: the RECON *deployer*

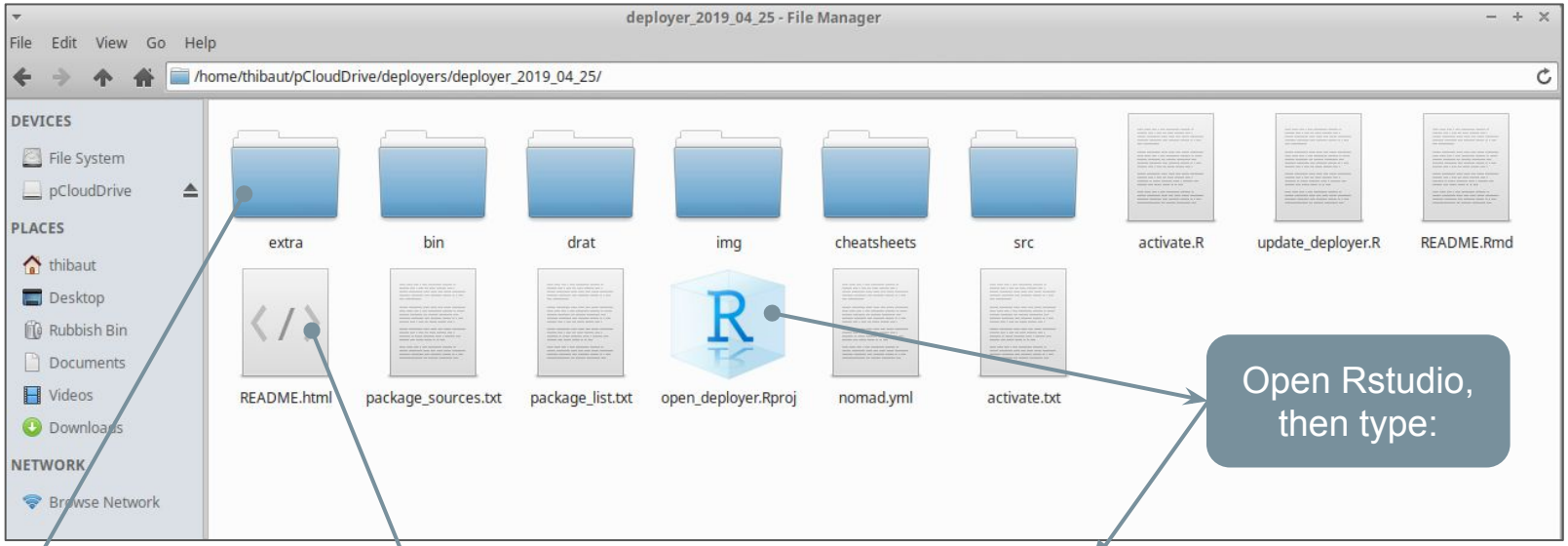
Taking R offline using the *deployer*



The RECON deployer

- USB stick with latest R, Rtools, Rstudio for Windows, MacOSX, Linux
- Local package repository - instance of *nomad*: https://github.com/reconhub/nomad
- ~2000-3000 CRAN packages
- ~10-20 github packages
- Cheatsheets
- Website: https://github.com/reconhub/deployer

Using the *deployer*



Software: R, Rstudio, git

Explanations & instructions

Open Rstudio, then type:

```
## activate the deployer
source("activate.R")

## install packages from the deployer
## note: xxx can be a non-CRAN package
install.packages("xxx")
```

To go further...

Resources for the *reportfactory*

- Website: https://github.com/reconhub/reportfactory
- Factories response Ebola DRC 2019 : https://github.com/reconhub/report_factories_templates
- R4epi templates: https://r4epis.netlify.com/

Resources for *linelist*

- Website: https://www.repidemicsconsortium.org/linelist/
- Github: https://github.com/reconhub/linelist

Resources for the *deployer*

- Github: https://github.com/reconhub/deployer
- *nomad*: https://github.com/reconhub/nomad