Open Computational Ecosystems and Reproducible Research

Markus Kainu, Joona Lehtomäki, Juuso Parkkinen, Juha Yrjölä, Måns Magnusson, Mikko Tolonen, Niko Ilomäki, Leo Lahti

Contact: http://ropengov.github.io

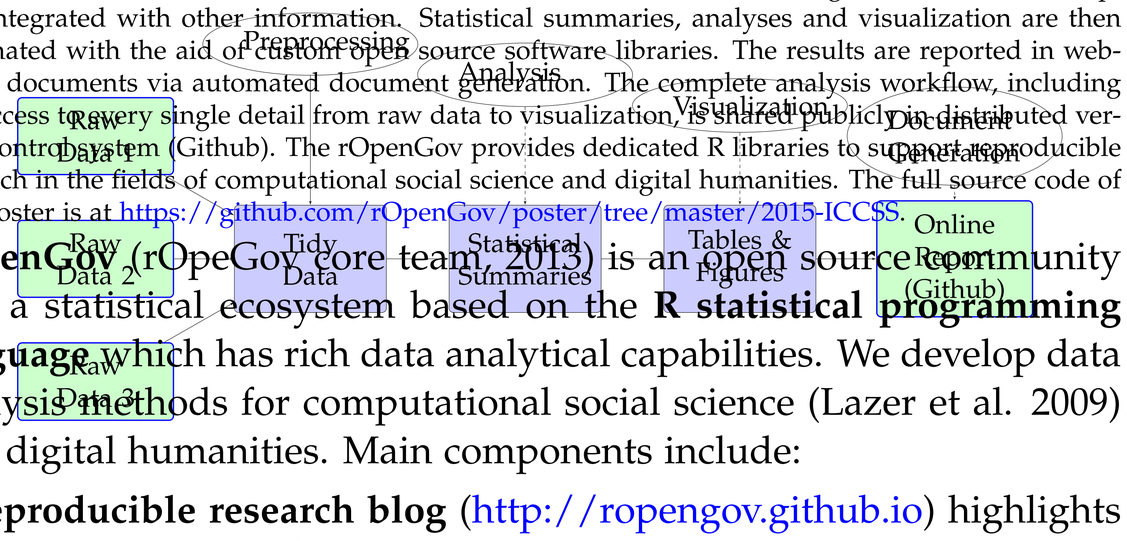
Open data analytics The recent explosion in open data availability has created novel opportunities for research. Efficient data analytical tools are crucial for taking full advantage of digital data streams. Custom software libraries are now rapidly emerging and have a huge potential to contribute to transforming computational social sciences, digital humanities, and related fields.

Advantages of the open development model Efficient data analysis relies on customized workflows that are best developed jointly by the user community, as already is the standard practice in bioinformatics (Bioconductor), particle physics, and other fields (rOpenSci). Similar communities are now shaping up in social sciences and humanities. The open analytics has many benefits (Ioannidis 2014, Morin et al. 2012):

- **Efficiency**: Many standard tasks can be automated, leaving researchers more time to focus on the specific problems.
- **Transparency**: Full details of the analysis from raw data to the final results are available.
- **Reproducibility**: Reproducible analyses can be repeated exactly without human intervention and modified easily.
- **Standardization**: A developer community can pool scarce research resources, develop standards in data analysis and ensure compatibility.
- **Open source**: Open licensing guarantees that the tools are freely available for the international scientific community. We use GitHub for shared version control and open development.

The map above shows *disposable incomes of private households at NUTS2 level regions in European Union in 2011*. This poster, including the Eurostat analysis example above, is fully reproducible. Download the full source code at https://github.com/ropengov/poster

Reproducible research workflow Raw data sets are downloaded from original sources, tidied up, and integrated with other information. Statistical summaries, analyses and visualization are then automated with the aid of custom open source software libraries. The results are reported in web-based documents via automated document generation. The complete analysis workflow, including full access to every single detail from raw data to visualization, is shared publicly in distributed version control system (Github). The rOpenGov provides dedicated R libraries to support reproducible research in the fields of computational social science and digital humanities. The full source code of this poster is at https://github.com/rOpenGov/poster/tree/master/2015-ICCSS.



rOpenGov (rOpeGov core team, 2013) is an open source community and a statistical ecosystem based on the **R statistical programming language** which has rich data analytical capabilities. We develop data analysis methods for computational social science (Lazer et al. 2009) and digital humanities. Main components include:

- **Reproducible research blog** (http://ropengov.github.io) highlights the opportunities of open data analytics.
- **Online tutorials** demonstrate how to access and analyse open data streams.
- **R packages** provide the means to share computational algorithms to support reproducible data analysis. Our collection includes tools for open data in various countries (Finland, Poland, Russia, USA), cities (Helsinki), statistics authorities (Eurostat, PX-Web, QOG), data anonymization, geographic information (OpenStreetMap, WFS), weather, demography, bibliograpies, media APIs, political science, elections and parliamentary monitoring. For a full list, see http://ropengov.github.io/projects

```r
download.file("http://ec.europa.eu/eurostat/cache/GISCO/geodatafiles/NUTS_2010_60M_SH.zip", destfile="NUTS_2010_60M_SH.zip")
unzip("NUTS_2010_60M_SH.zip")
df <- get_eurostat("tgs00026", time_format = "raw")
library(eurostat)
library(tidyr)
df$time <- eurotime2num(df$time)
df <- df[df$time == max(df$time),]
library(rgdal)
map <- readOGR(dsn = "./NUTS_2010_60M_SH/Data", layer = "NUTS_RG_60M_2010", verbose = FALSE)
map_nuts2 <- subset(map, STAT_LEVL_ == 2)
NUTS_ID <- as.character(map_nuts2$NUTS_ID)
VarX <- rep(NA, 316)
dat <- data.frame(NUTS_ID,VarX)
dat2 <- merge(dat,df,by.x="NUTS_ID",by.y="geo", all.x=TRUE)
row.names(dat2) <- dat2$NUTS_ID
row.names(map_nuts2) <- as.character(map_nuts2$NUTS_ID)
## order data
dat2 <- dat2[order(row.names(dat2)), ]
map_nuts2 <- map_nuts2[order(row.names(map_nuts2)), ]
## join
library(maptools)
dat2$NUTS_ID <- NULL
shape <- spCbind(map_nuts2, dat2)
library(ggplot2)
library(rgeos)
shape$id <- rownames(shape@data)
map.points <- fortify(shape, region = "id")
map.df <- merge(map.points, shape, by = "id")
map.df$unit <- as.character(map.df$unit)
library(ggplot2)
library(scales)
library(grid)
p <- ggplot(data=map.df, aes(long,lat,group=group))
#p <- p + geom_polygon(data = map.df, aes(long,lat),fill=NA,colour="white",size = .2)
p <- p + geom_polygon(aes(fill = values), color= alpha("white", 1/2),  size=.2)
p <- p + coord_map(project="orthographic", xlim=c(-10,45), ylim=c(25,90))
p <- p + labs(title = paste0("Disposable household incomes in ",max(df$time)))
p <- p +  theme(legend.position = c(0.06,0.42),
                        legend.justification=c(0,0),
                        legend.key.size=unit(20,'mm'),
                        legend.direction = "vertical",
                        legend.background=element_rect(colour=NA, fill=alpha("white", 2/3)),
                        legend.text=element_text(size=30),
                        legend.title=element_text(size=30),
                        title=element_text(size=30),
                        panel.background = element_blank(),
                        plot.background = element_blank(),
#                         panel.grid.minor = element_line(colour = 'Grey80', size = .5, linetype = 'solid'),
#                         panel.grid.major = element_line(colour = 'Grey80', size = .5, linetype = 'solid'),
                        panel.grid.minor = element_blank(),
                        panel.grid.major = element_blank(),
                        axis.text = element_blank(),
                        axis.title = element_blank(),
                        axis.ticks = element_blank(),
                        plot.margin = unit(c(-3,-1.5, -3, -1.5), "cm"))
p <- p + guides(fill = guide_legend(title = "Euro per Year",
                                    title.position = "top",
                                    title.hjust=0))
p
```

References

1. J. Ioannidis (2014). How to Make More Published Research True? PLoS Medicine 11(10): e1001747.
2. D. Lazer et al. (2009). Computational Social Science 323, 721–723
3. A. Morin et al. (2012). Research priorities. Shining light into black boxes. Science 336, 159-160.
4. rOpenGov core team (2013). R ecosystem for open government data and computational social science. NIPS Machine Learning Open Source Software workshop (MLOSS). December 2013, Lake Tahoe, Nevada, US
   Political Science Review, 107(02), 326–343

We are thankful for a number of developers. For a full list, see http://ropengov.github.io