A generic shiny/js dashboard for data validation results

Statistics Netherlands

Olav ten Bosch and Mark van der Loo

UROS, Bucharest, 21-05-2019

Contents

- The ValidatFOSS project
- A generic validation report for the ESS
- Dashboard for data validation results
- Validation workflow in Shiny
- Wrap up

The validatFOSS project (1)

- Validation with ***F***ree and ***O***pen ***S***ource ***S***oftware
- Part of ***European validation projects*** together with Sweden, Portugal, UK, France, Poland, Hungary, Iceland, Italy, Lithuania, Estonia and Eurostat
- ***International validation:*** NSI -> Eurostat
  preventing "***data ping pong***"
- ***National validation:*** within NSI
  validation principle: "***the sooner the better***"
- Facilitate sharing of open source validation ***software*** and international validation ***rules***

The validatFOSS project (2)

Exisiting tools:

R-package ***validate***:

- Implements concepts of the *ESS handbook on validation*
- Supports rules that are per-field, in-record, cross-record or cross-dataset
- On CRAN and awesome list 

R-package ***validatetools***:

- Functions for finding *redundancies* or *contradictions*
- On CRAN and awesome list 

The validatFOSS project (3)

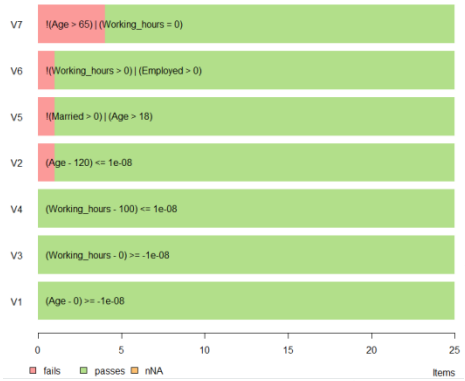
```
# Range limits:
Age >= 0
Age <= 120
Working_hours >= 0
Working_hours <= 100

# Some checks between variables:
if (Married > 0) Age > 18
if (Working_hours > 0) Employed > 0

#Such a rule depends on country legislation:
if (Age > 65) Working_hours = 0

# ID must be unique
any(duplicated(ID)) == FALSE
```

ID	Age	Marital status	Status in employment	Working hours per week
1	36	0	1	40
2	40	1	1	40
3	25	0	0	0
4	31	0	1	20
5	62	1	1	43

Validate
confront

```
> summary(validation)
  name items passes fails nNA error warning                                expression
1   V1    25     25     0   0 FALSE   FALSE                                (Age - 0) >= -1e-08
2   V2    25     24     1   0 FALSE   FALSE                              (Age - 120) <= 1e-08
3   V3    25     25     0   0 FALSE   FALSE                  (Working_hours - 0) >= -1e-08
4   V4    25     25     0   0 FALSE   FALSE                (Working_hours - 100) <= 1e-08
5   V5    25     24     1   0 FALSE   FALSE                   !(Married > 0) | (Age > 18)
6   V6    25     24     1   0 FALSE   FALSE !(Working_hours > 0) | (Employed > 0)
7   V7    25     21     4   0 FALSE   FALSE     !(Age > 65) | (Working_hours = 0)
8   V8     1      0     1   0 FALSE   FALSE          any(duplicated(ID)) == FALSE
```

Results



The validatFOSS project (4)

- Apply FOSS tools in multiple ***domains***:
  - ***Short term statistics*** (STS): 10 internationally agreed validation rules in R-validate code
  - ***National Accounts*** (NA): Additivity rules, price checks and cross table rules from ESA 2010
  - ***Generic rules***: identified by ESTAT in VTL 2.0
  - ***Energy statistics***
  - ***Tourism statistics***

The ValidatFOSS project (5)

2 STS validation rules in R-validate syntax:

```
- expr: if (INDICATOR %in% c("IMPZ","PRBB","PREN","PREX","PREZ","PRIN","PRON")) OBS_VALUE > 0
  name: "STS03"
  label: "Prices positive"
  description: "Zeroes are not admitted for prices."
  created: 2019-03-01 15:41:02
  origin: rules.R
  meta: []

- expr: anyDuplicated(.[names(.) != "OBS_VALUE"]) == FALSE
  name: 'STS05'
  label: 'unique observations'
  description: |
    Different values for the same observation (double values)
    are not accepted in one file.
  created: 2019-03-01 15:41:02
  origin: rules.R
  meta: []
```

A generic validation report for the ESS (1)

Validation systems and languages differ in the ESS:

```
if employment status == "old-age pensioner" and
age < 35  then error "Too young!"
0.5 < turnover(curMonth)/turnover(prevMonth) < 2
```

Lithuania

Germany

```
WENN ANZAHL VON Familie[ALLE].Person[MIT Alter < 18] > 0 DANN ... ENDE
```

```
IF maritalstate=married THEN
  Age<=15 "Too young to be married"
ENDIF
```

Blaise (NL)

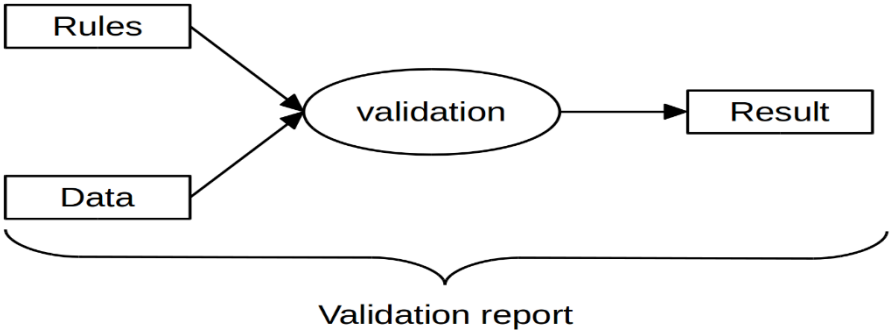
```
profit <= 0.6*revenue
```

R-validate (NL)

A generic validation report for the ESS (2)

ESSnet 2018 designed a generic validation report for:

- ***Every*** statistical ***domain***
- ***Every*** statistical validation ***tool***
- For ***ESS*** data ping pong as well as ***within NSI's***
- For ***micro***data as well as ***aggregated*** data



[Standard ESS validation report](#) (pdf)

A generic validation report for the ESS (3)

Example (JSON):


```
{
  "event": {
    "time": "20170518T105055+02",
    "actor": "R 3.4.0",
    "agent": null,
    "trigger": null
  },
  "rule": {
    "language": "R pkg validate 0.1.7",
    "expression": "income >= 0",
    "severity": "error",
    "description": "total income must be non-negative"
  },
  "data": [
    "Dutch inhabitants",
    "Household survey 2017",
    "8237193679",
    "Household Income"
  ],
  "value": "1"
}
```


A generic validation report for the ESS (4)

R-package ***validatereport***:

- Implements the *validation report standard*

```
# load data and rules:
dat <- read.csv('data/Task2_Data.csv')
rules <- validator(.file="data/Task2_Rules.R")

# Confront data with rules:
validation <- confront(dat, rules, key="ID")

# Generate the report:
export_ess_validation_report(validation, rules, file="Task2_Report.json")
```

github.com/data-cleaning/validatereport

Dashboard for data validation results (1)

Validation report from any system


```
[
  {
    "rule": {
      "expression": "check(DS.hours_worked between 1 and 80)",
      "severity": "warning"
    },
    "event": {
      "time": "2017-09-01T07:51:44.933Z",
      "actor": "Eurostat"
    },
    "data": {},
    "value": "0" // failed
  },
  {
    "rule": {
      "expression": "cost + profit == turnover",
      "severity": "error"
    },
    "event": {
      "time": "2017-09-01T07:51:46.933Z",
      "actor": "Eurostat"
    },
    "data": {},
    "value": "1" // passed
  },
  ...
  ...
  ...
]
```


Dashboard

- Viewing
- Filtering
- Aggregation

[DEMO]

markdown

html pdf word

Dashboard for data validation results (2)

Concept:

- Show validation results in ***data context***
- ***Viewing***, ***filtering***, ***aggregation*** of validation results
- Filters on fails/passes, severity, per rule, per data cell

Architecture based on standard components:

- ***Crossfilter.js*** (https://crossfilter.github.io/crossfilter): exploring large multivariate datasets in the browser
- ***dc.js*** (https://dc-js.github.io/dc.js): dimensional charting
- ***Datatables***: viewing data in colored datagrid

Validation workflow in Shiny (1)

unconfUROS 2018: ***Validaty***

- Shiny dashboard for 'validate‘ and ‘validatetools’ github.com/uRosConf/validaty

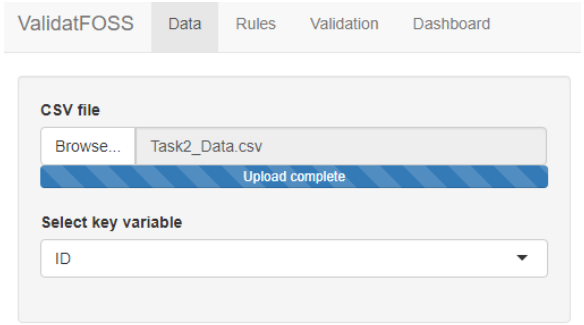
Challenge: ***web*** versus ***Shiny***

- Web version: usable by ***any*** (non-R) validation tool
- Shiny: better ***integration*** with R validation packages

Choice:

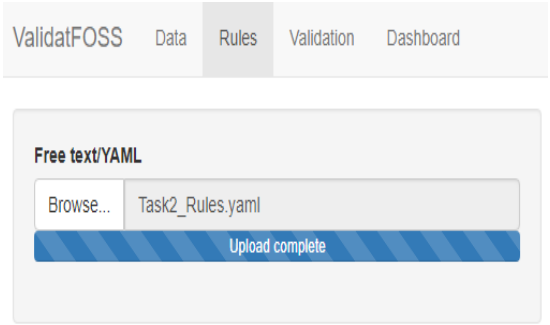
- JavaScript Dashboard => ***htmlWidget***
- Validaty + dashboard => ***validation workflow***

Validation workflow in Shiny (2)



Choose data

=>



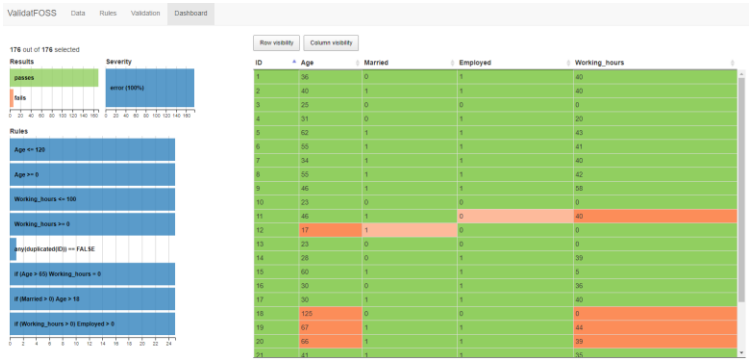
Choose rules

=>

Demo

Validate

=>



Explore

Wrap up

- Generic R validation tools for ***international*** as well as ***national*** validation in the ESS
- A ***generic validation report*** for the ESS
- R package ***validatreport*** creates such generic report
- Validation ***dashboard*** facilitates ***viewing***, ***filtering***, ***aggregation*** of validation results
- Shiny ***validation workflow*** based on validaty + dashboard:

  load data => rules => validate => explore results

Questions, ideas, suggestions

?

Olav ten Bosch
Mark van der Loo
obos@cbs.nl
mplo@cbs.nl

Curated list of software for official statistics

www.awesomeofficialstatistics.org