An R-based data editing system

Mark van der Loo

Statistics Netherlands

R at Statistics Netherlands

- Strategic tool since 2010
  - Internal wiki, knowledge center, course.
- Used at:
  - National Accounts
  - Tourist statistics
  - Data collection with web robots (part of CPI)
  - Computing HSMR
  - Derivation of households
  - *Etc. etc. etc.*
- Used for:
  - (Complex) data manipulation
  - Analyses and regression
  - Visualisation
  - Data editing

Packages developed, at CRAN

- Data editing
  - editrules
  - deducorrect
  - rspa
- Data Visualisation
  - treemap
  - tabplot, tabplotd3
- Large data files
  - LaF (Large ASCII files)

Data editing packages

- editrules
  - Define rules
  - Verify data against them
  - Localize errors
- deducorrect
  - Deductive correction
  - Deductive imputation
  - Apply ‘direct rules’
- rspa
  - Adjust numerical records to satisfy rules

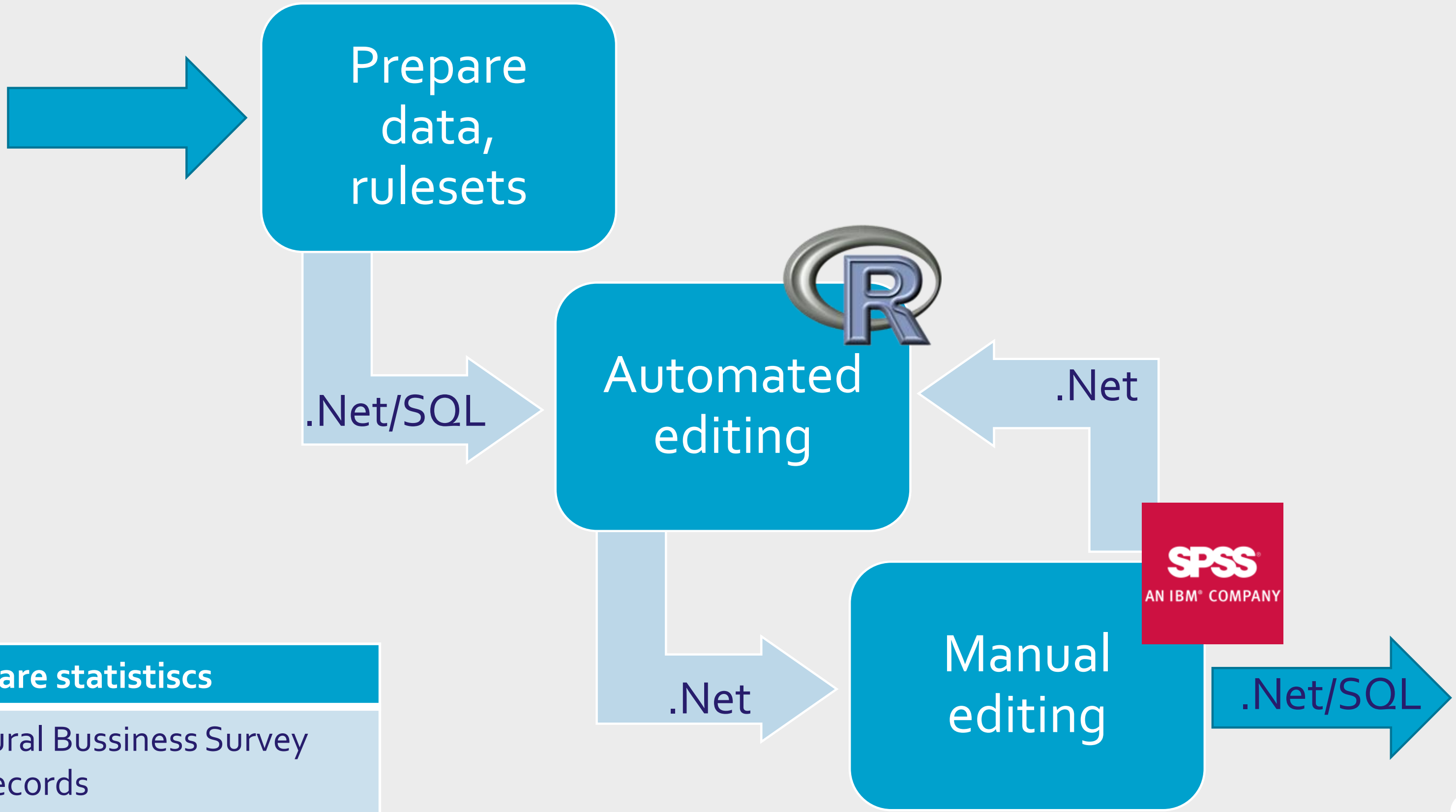
```
> E <- editfile(‘myrules.txt’)
> ve <- violatedEdits(E,dat)
> el <- localizeErrors(E,dat)
>
```

```
> E <- editfile(‘myrules.txt’)
> ct <- correctTypos(E,dat)
> cr <- correctRounding(E,dat)
>
```

```
> E <- editfile(‘myrules.txt’)
> ad <- adjustRecords(E,dat)
>
```

Rules defined with ‘editrules’ are reused by `deducorrect’ and `rspa’

Automated data editing system for Child Care Centre Statistics



Prepare data, rulesets

.Net/SQL

Automated editing

.Net

.Net

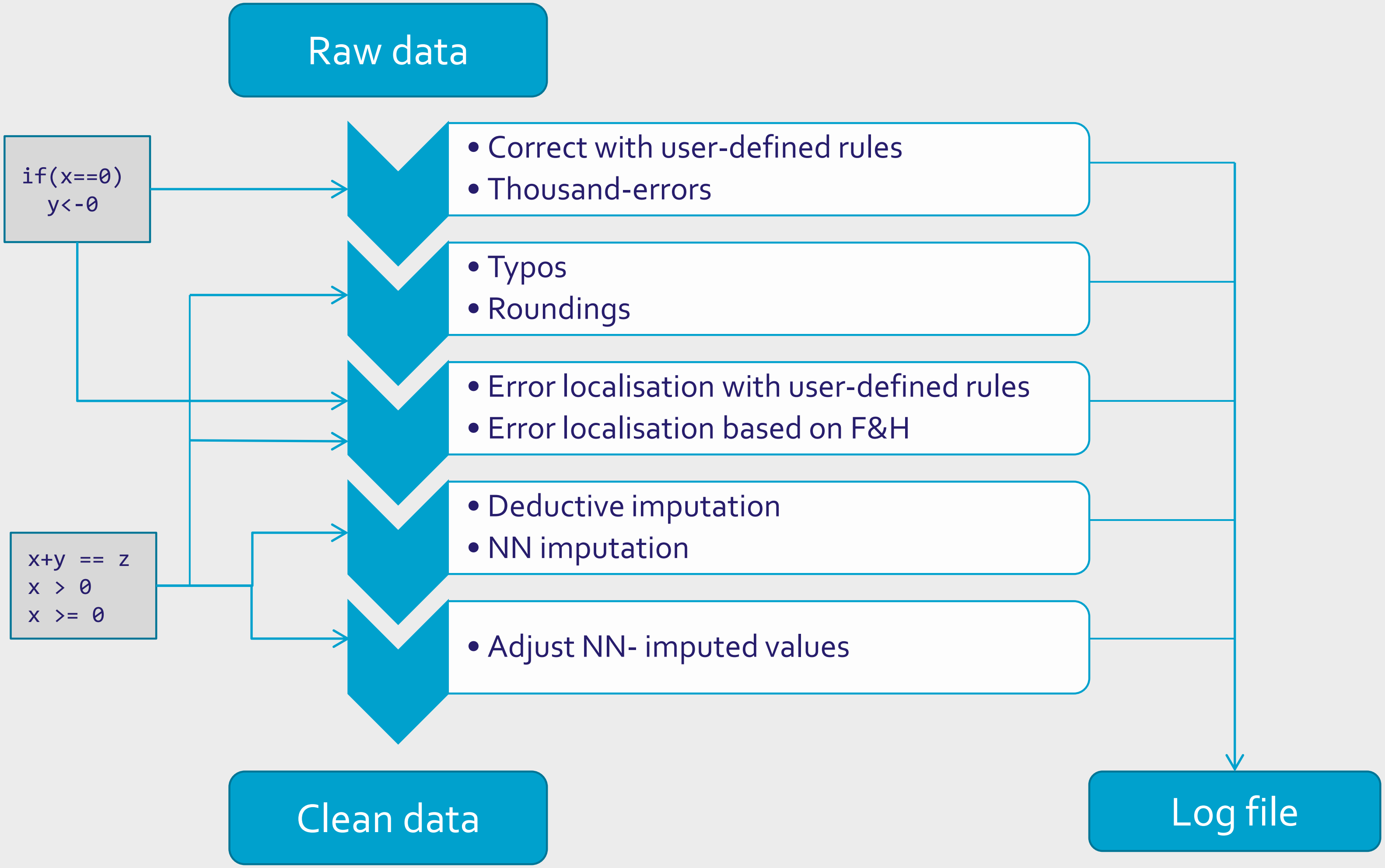
Manual editing

.Net/SQL

Child care statistiscs

Structural Bussiness Survey
~800 records
~80 linear rules (balance edits)
~50 variables

Automated editing



Raw data

```
if(x==0)
  y<-0
```

- Correct with user-defined rules
- Thousand-errors

- Typos
- Roundings

- Error localisation with user-defined rules
- Error localisation based on F&H

- Deductive imputation
- NN imputation

- Adjust NN- imputed values

```
x+y == z
x > 0
x >= 0
```

Clean data

Log file

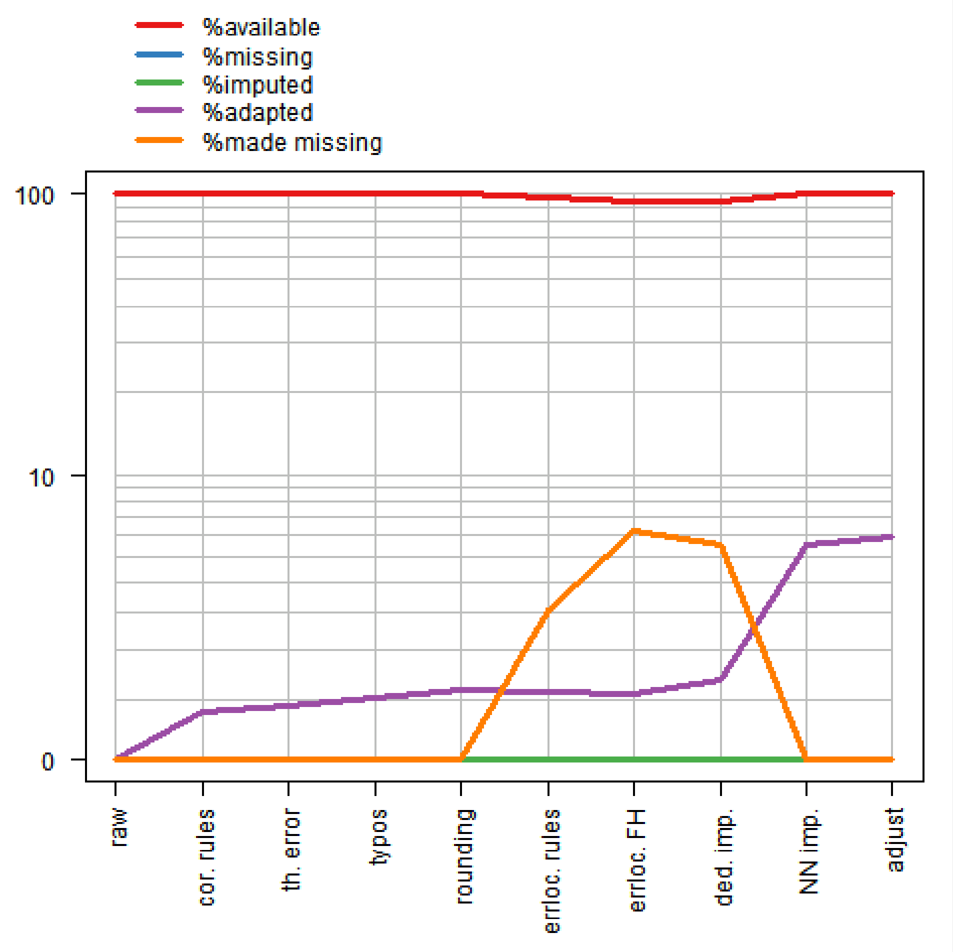
Example code: solve typing errors

```
oplossenTikfouten <- function(E, dat, db, id){
   d <- correctTypos(E,dat)
   cors <- d$corrections
   opslaanLogRecords(db,
      id        = dat[cors$row,id],
      variabele = cors$variable,
      oud       = cors$old,
      nieuw     = cors$new,
      methode   = "tikfout"
   )
   d$corrected
}
```

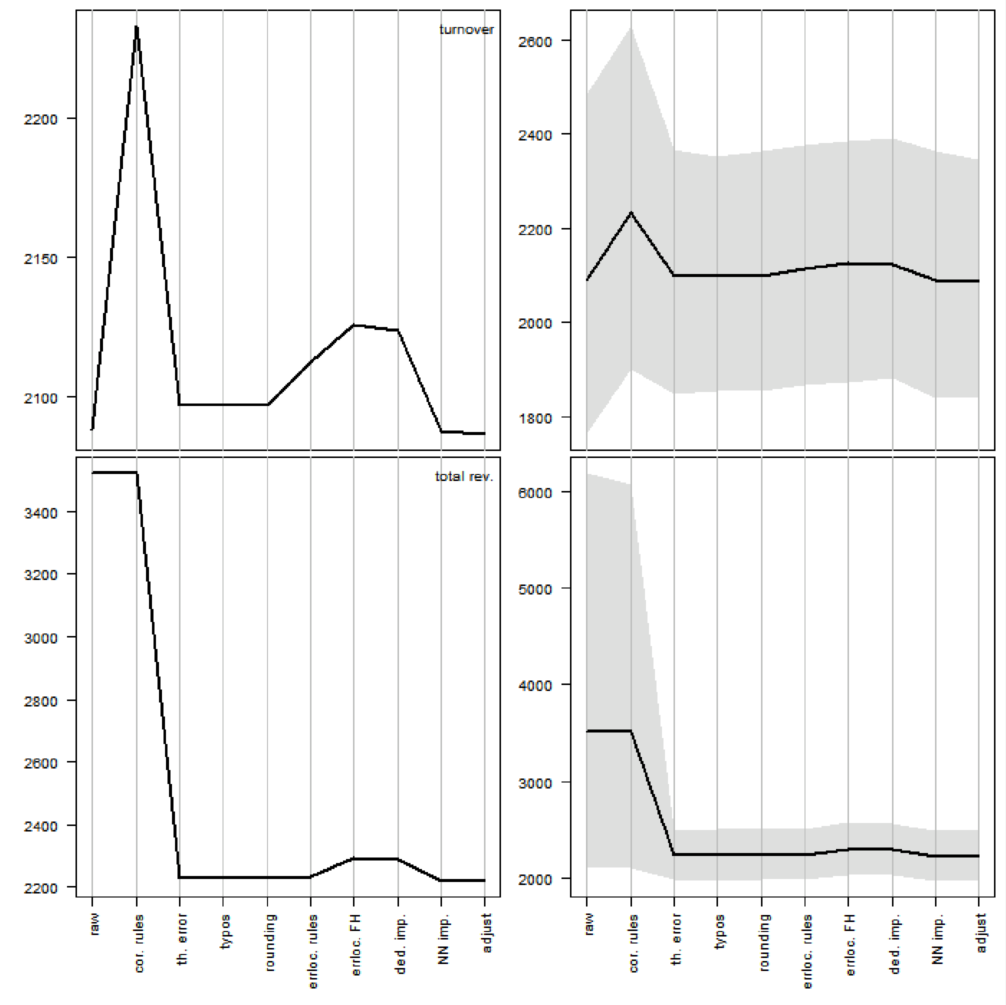
Results and process flow I: Cells

Cells				
Available			Missing	
Still available		Imputed	Made missing	Still missing
Available unadapted	Available adapted			

Results and process flow I: Cells



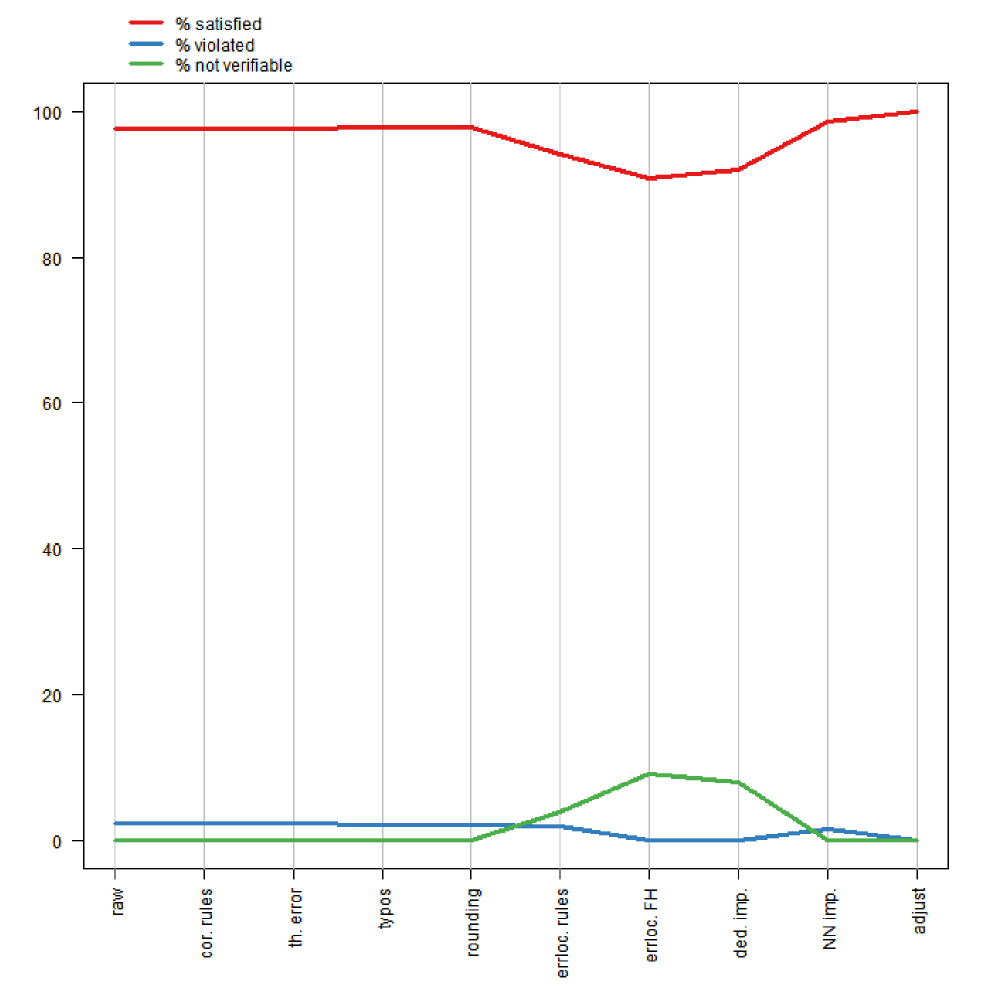
Results and process flow II: Aggregates



Results and process flow III: violations

Nr of checks: #Rules X #Records					
Verifiable				Not verifiable	
Violated		Satisfied			
Still violated	Extra violated	Still satisfied	Extra satisfied	Still not verifiable	Extra not verifiable

Results and process flow III: violations



Results and process flow IV: measure of violation

An edit rule *e* can be understood as a 3-valued function of a record ***x***:

$$e(\boldsymbol{x}) = \begin{cases} 1\ if\ x\ satifies\ e \\ 0\ if\ x\ violates\ e \\ NA\ if\ e\ (\boldsymbol{x}) cannot\ be\ determined \end{cases}$$

Tolerance: how much do I need to change ***x*** *so e(****x****)=1?*

Results and process flow IV: measure of violation (single rule)

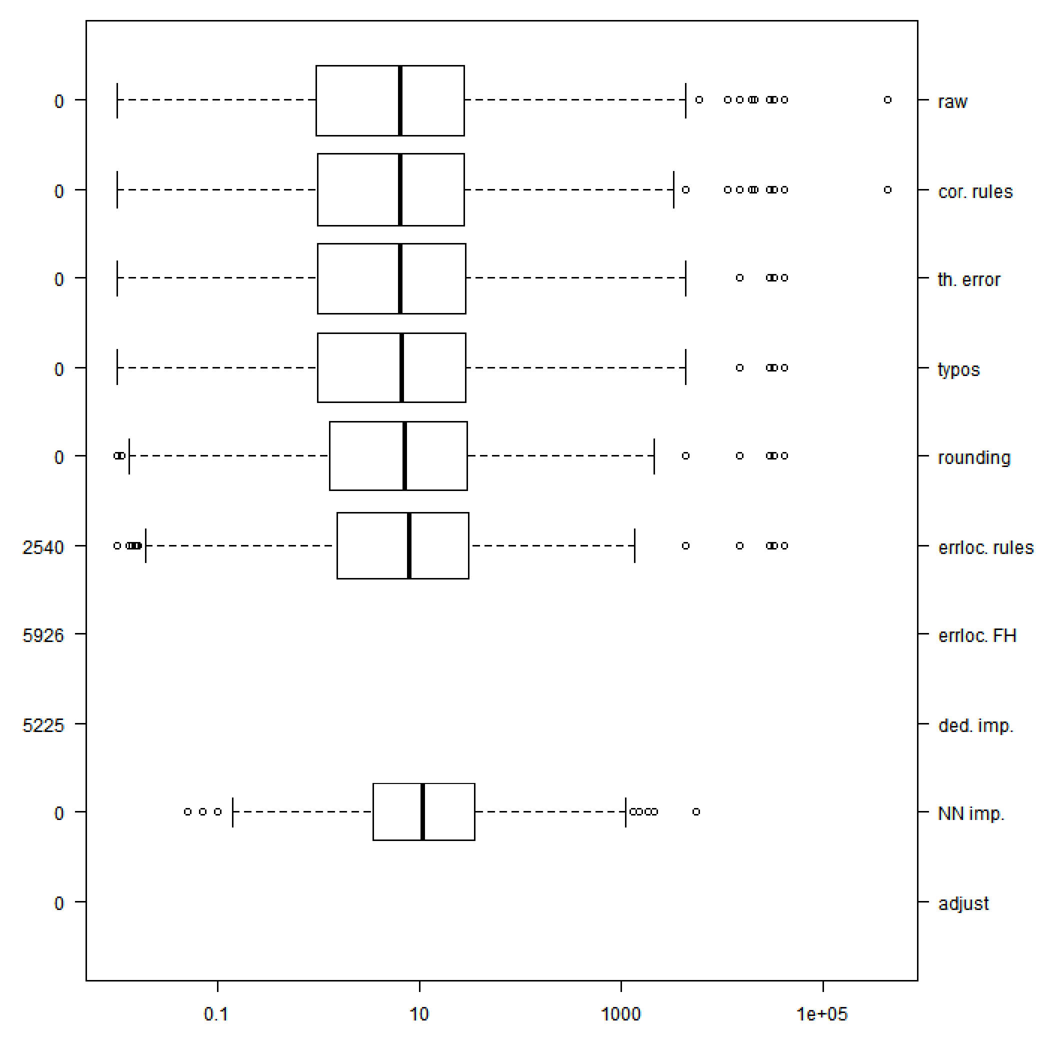
An edit rule *e* can be understood as a 3-valued function of a record ***x***:

$$e(\boldsymbol{x}) = \begin{cases} 1\ if\ x\ satifies\ e \\ 0\ if\ x\ violates\ e \\ NA\ if\ e\ (\boldsymbol{x})\ cannot\ be\ determined \end{cases}$$

Tolerance: $\underbrace{\textit{how much}}_{\text{Euclidean distance}}$ *do I need to change* $\underbrace{\boldsymbol{x}}_{\text{numeric}}$ *so* $\underbrace{e(\boldsymbol{x})}_{\text{linear}}$*=1?*

In this case there is a closed-form solution

Results and process flow IV: measure of violation (single rule)



Positive tolerances per rule

Height of box ~ square root of nr of violations

Left axis denotes nr of unevaluated rules.

Results and process flow IV: measure of violation (multiple rules)

Given a set of rules $e_1, e_2, \dots, e_n$ that a record ***x*** must obey.

How much do I need to change **x***, so that all* $e_i(\boldsymbol{x}) = 1$*?*

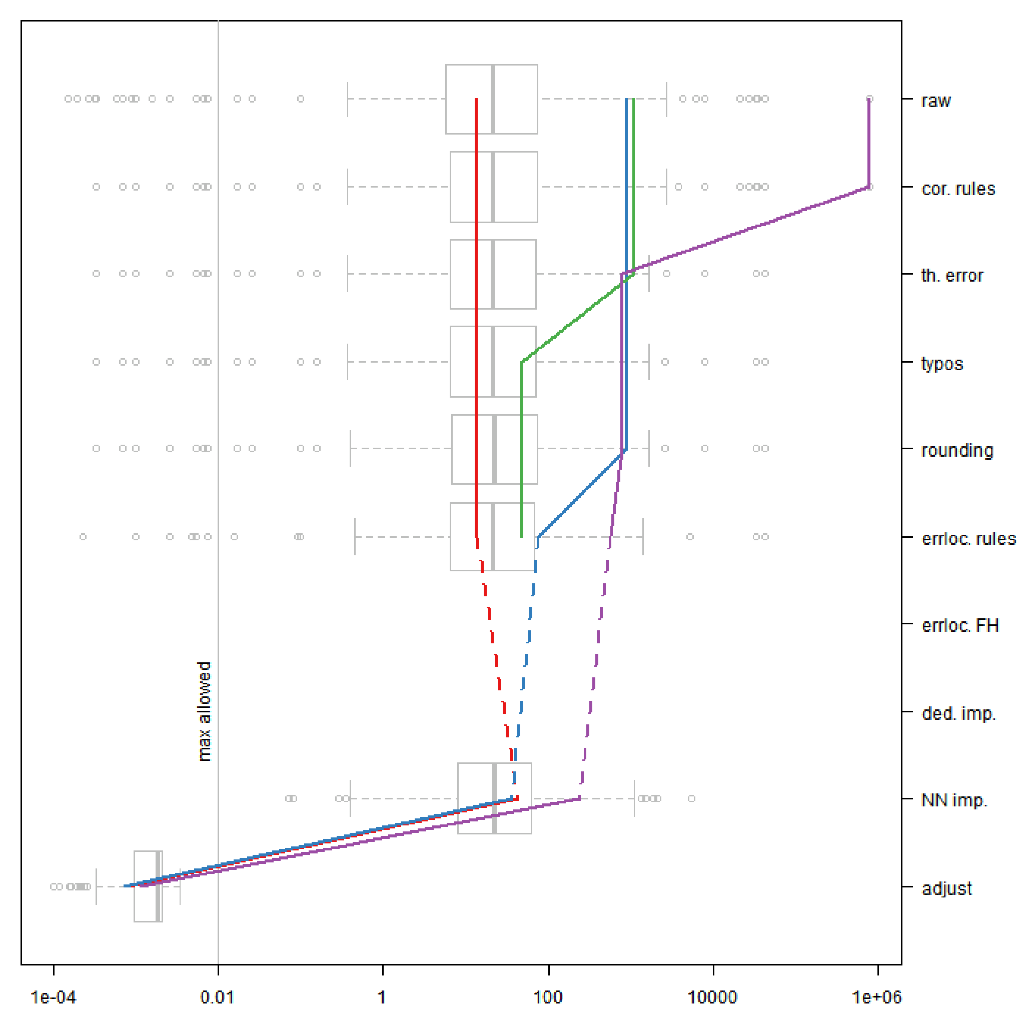
Results and process flow IV: measure of violation (multiple rules)

Given a set of rules $e_1, e_2, \ldots, e_n$ that a record $\boldsymbol{x}$ must obey.

$$\underbrace{\textit{How much}}_{\text{Euclidean distance}}\ \textit{do I need to change}\ \underbrace{\mathbf{x},}_{\text{numeric}}\ \textit{so that}\ \underbrace{\textit{all}\ e_i(\boldsymbol{x})}_{\text{linear}} = 1?$$

This can be computed with the rspa package.

Results and process flow IV: measure of violation (multiple rules)



Euclidean distance between actual and closest valid record.

A line traces one record.

Conclusions and outlook

- R-based, easy to build production-grade data editing system.
- Logging and indicators offer insight into
  - Quality of automated cleaning
  - Quality of data
- Future plans:
  - System is now being configured for another statistic
  - Implement general indicators (validator package)
  - Separate logging stream from data stream
- Reference
  - E. de Jonge and M. van der Loo *An introduction to data cleaning with R* (SN discussion paper nr 201313)