

# Finding and handling data errors

## errorlocate

Edwin de Jonge and Mark van der Loo

CBS, Department of Methodology

uRos2019 Tutorial Session, Bucharest

# Error localization

Data validation and error localization answer different questions.

## **Data validation**

Which errors are there?

## **Error localization**

Where do I need to make changes to fix the errors?



# Example

## Ruleset

```
age >= 0  
age <= 120  
if (drivers_licence == TRUE) age >= 18
```

## Data

age	drivers_licence
10	TRUE

## Question:

Which field or fields would you change?

# Error localization

## Definition

Error localization is a procedure that points out fields in a data set that can be altered or imputed in such a way that all validation rules can be satisfied.

# Example

## Ruleset

```
if (married == TRUE ) age >= 16  
if (attends == "kindergarten") age <= 6
```

## Data

age	married	attends
3	TRUE	kindergarten

## Question

Which field or fields would you change?



# Principle of Fellegi and Holt

Find the minimal (weighted) number of fields to adjust such that all rules, including implied rules, can be satisfied.

IP Fellegi and D Holt, JASA **71** 353 17–35 (1976).

## Note

This should be used as a last resort, when no further information on the location of errors is available.



# Implied rules?

```
turnover - total.cost == profit  
profit <= 0.6 * turnover
```

This implies (substituting profit):

```
total.cost >= 0.4 * turnover
```

We need to take into account such *essentially new* rules (edits) —unstated relations between variables that can be derived from the explicitly defined rules.



## Data example 2

```
age >= 0,  
age < 150,  
if (driver_license == TRUE) age >= 16
```

age	driver_license
10	TRUE

### Question

Which variable would you adjust?

**It depends on the quality of age and driver\_license. We can add more weight to age if we think that variable has better quality.**





## Assignment:

```
rules <- validator(  
  if (married == TRUE) age >= 16,  
  if (attends == "kindergarten") age <= 6  
)  
  
persons <- data.frame( age      = 3  
                      , married = TRUE  
                      , attends  = "kindergarten"  
                      )
```

- check with `validate` which rules are violated.
- What should be changed to this record to “correct” it? Why?



# Feligi Holt (FH) formalism:

*Find the minimal (weighted) number of variables that cause the invalidation of the data rules.*

Makes sense if no further knowledge on the error mechanism is available!  
R package `errorlocate` (second generation of `editrules`) implements FH.

# Feligi Holt (FH) formalism:

But there are exceptions. . .

- In balance sheets, swapping variables (2 edits) sometimes makes more sense than adjusting one value (1 edit). (see R package:deducorrect).
- In some data, spreading a surplus or shortage on a variable over many variables is sensible. (see R package: rspa).



# errorlocate

- R-package that implements FH.
- Is extensible (you can plug in your own detection stuff)
- provides:
  - `locate_errors`
  - `replace_errors`
  - R5 classes to add your own stuff.

## errorlocate::locate\_errors

```
locate_errors( data.frame( age  = 3
                           , married = TRUE
                           , attends = "kindergarten"
                           )
  , validator( if (married == TRUE) age >= 16
               , if (attends == "kindergarten") age <= 6
               )
  )
```

```
## call: x$locate(data = data, weight = weight, ...)
```

```
## located 1 error(s).
```

```
## located 0 missing value(s).
```

```
## Use 'summary', 'values', '$errors' or '$weight', to explore and retrieve
```



## errorlocate::locate\_errors

```
locate_errors( data.frame( age  = 3
                           , married = TRUE
                           , attends = "kindergarten"
                           )
  , validator( if (married == TRUE) age >= 16
               , if (attends == "kindergarten") age <= 6
               )
  )$errors
```

```
##           age married attends
## [1,] FALSE      TRUE  FALSE
```



## Assignment (small examples)

a) Find the error in this record with `locate_errors`:

age	married	attends
26	TRUE	kindergarten

b) Find the error with `locate_errors`:

age	married	attends
15	TRUE	kindergarten



# Removing errors

- Detecting errors is very useful, but then what?
- Fixing philosophy is:
  - Find erroneous values.
  - Remove them (i.e. make them NA).
  - Impute them with sensible values.

## Note

We could also remove erroneous records completely, but often this result in *over-deletion* and introduces a *bias*.





## errorlocate::replace\_errors

- Locates errors and replaces them with NA.

```
replace_errors(  
  data.frame( age      = 3  
              , married = TRUE  
              , attends = "kindergarten"  
            )  
  , validator( if (married == TRUE) age >= 16  
              , if (attends == "kindergarten") age <= 6  
            )  
)
```

```
##   age married    attends  
## 1    3      NA kindergarten
```



# Internal workings:

`errorlocate:`

- translates error localization problem into a **mixed integer problem**, which is solved with `lpsolveAPI`.
- contains a small framework for implementing your own error localization algorithms.

# Pipe friendly

The `replace_errors` function is pipe friendly:

```
rules <- validator(age < 150)

data_noerrors <-
  data.frame(age=160, driver_license = TRUE) %>%
  replace_errors(rules)

errors_removed(data_noerrors) # contains errors removed
```

