

TALOS eudract AE reporting - example

AGDamsbo

Knitted: 12 November 2021

Contents

Intro	1
Data import	2
Modified trial-specific adjudication list with added maddra codes from eudract	2
Formatting to EUDRACT	2
Adding columns according to safety-format	3
Fatal outcome	3
Related	3
Serious	3
Randomisation	4
SOC kode og term/subcat	4
Converting to new, long data.frame	5
Matching adj_code to soc_code	6
Configuring XML	6
Creating specified data frame	6
Handling Deaths	7
Creating safety summary	7

Intro

This is the data management workflow for the reporting of AEs in the TALOS trial to the EudraCT database. A dedicated package is used to format data after a longer process of editing data to conform. The source data is not perfectly formatted, and during the process, a few manual steps are necessary. - Advise number one: Make sure to format data according to the desired format for reporting.

This is “page” 2 of 2. All cleaning is performed in “TALOS AE cleaning.Rmd”

```
setwd("/Volumes/Data/TALOS/")
```

```
library(haven)
library(dplyr)
# https://www.rdocumentation.org/packages/eudract/versions/0.9.3
library(eudract)
```

Data import

Data set

```
d<-as_factor(read_dta("/Volumes/Data/TALOS/talos_ae_clean.dta"))
write.csv(head(d,100),"sample_ae.csv")
d<-read.csv("sample_ae.csv")
```

Modified trial-specific adjudication list with added maddra codes from eudract

```
library(readxl)
adj_tbl<-read_xlsx("adjudication_table.xlsx")
# write.csv(soc_code,"soc_code.csv") # Export af soc_code til manuel kodning af oprindelig Adjudication
head(adj_tbl)
```

```
## # A tibble: 6 x 6
##   code subcat                                cat      soc_term  eutctId meddra
##   <chr> <chr>                                <chr>    <chr>      <dbl> <dbl>
## 1 000   Disabling Stroke - Haemorrhagic        Neurov~ Nervous s~ 1.00e11 1.00e7
## 2 001   Disabling Stroke - Ischemic            Neurov~ Nervous s~ 1.00e11 1.00e7
## 3 002   Intracerebral Haemorrhage (Non-Stroke) Neurov~ Nervous s~ 1.00e11 1.00e7
## 4 003   Non-Disabling Stroke - Haemorrhagic    Neurov~ Nervous s~ 1.00e11 1.00e7
## 5 004   Non-Disabling Stroke - Ischemic        Neurov~ Nervous s~ 1.00e11 1.00e7
## 6 005   Subarachnoid hemorrhage (SAH)          Neurov~ Nervous s~ 1.00e11 1.00e7
```

```
head(soc_code)
```

```
##
##           soc_term eutctId  meddra
## 1   Blood and lymphatic system disorders 1e+11 10005329
## 2           Cardiac disorders 1e+11 10007541
## 3 Congenital, familial and genetic disorders 1e+11 10010331
## 4           Ear and labyrinth disorders 1e+11 10013993
## 5           Endocrine disorders 1e+11 10014698
## 6           Eye disorders 1e+11 10015919
```

Formatting to EUDRACT

```
# Included data example in the eudract-package
head(safety)
```

```
##      pt  subjid related      soc fatal serious      group
## 1 10000081 US6-006  FALSE 10017947    0      1    Control
## 2 10000081 N04-006  FALSE 10017947    0      0  Experimental
## 3 10000891 US3-002   TRUE 10007541    0      0  Experimental
## 4 10002383 US6-012  FALSE 10007541    0      0  Experimental
## 5 10002895 US5-001  FALSE 10047065    0      0  Experimental
## 6 10002916 US8-006  FALSE 10042613    0      1    Control
##
##          term
## 1      Abdominal pain
## 2      Abdominal pain
## 3 Acute myocardial infarction
## 4      Angina pectoris
## 5      Aortic dissection
## 6      Aortic valve replacement
```

Adding columns according to safety-format

Fatal outcome

The “dstatus” contains final status of every event instance, with those marked Dødelig being used. The other option would be to code if the event was GCP monitored and is used.

```
# for (i in 1:nrow(d)){
#   d$fatal[i]<-ifelse(str_contains(d$description[i],adj_tbl$code[adj_tbl$soc_term=="Death"]),logic = "o
# }

d$fatal<-ifelse(d$status=="Dødelig",1,0)
table(factor(d$fatal),factor(d$rtreat))
```

```
##
##      Aktiv (Citalopram) Placebo
## 0          53          47
```

Related

All events coded with either of the three categories a considered related in this binary form.

```
d$related_bin<-ifelse(d$related=="Mulig"|
                      d$related=="Sandsynlig"|
                      d$related=="Afgjort relateret",
                      1,0)
```

Serious

Only SAEs are occurring, no SAR or SUSAR

```
ser<-c("SAE", "SAR", "SUSAR")
d$serious<-ifelse(d$CLFint %in% ser,1,0)
```

Randomisation

Group naming according to groups defined on the EudraCT page.

```
d$group<-ifelse(d$rtreat=="Placebo", "Placebo", "Active")
```

SOC kode og term/subcat

```
ls<-list()
for (i in 1:nrow(d)){
  # Text string split at ":", " ", "+" or "(" and constrained to first three digits.
  # The last step as a security against a missing " " following the adjudication code or similar.
  v<-substr(unlist(strsplit(d$description[i], "[: +()")"),1,3)
  # vector elements contained in adj_tbl$code are subset and added to list
  ls[[paste0("index", i)]] <- grep(paste(adj_tbl$code, collapse="|"), v, value = TRUE)
}
```

Splitting each list element into different columns, length(ls) equals nrow(d)

```
for (i in 1:length(ls)){
  # Subsets liste efter navngivning i forrige loop
  v<-ls[[paste0("index", i)]]
  for (j in 1:length(v)){
    # Føjer til eksisterende, tilføjer ekstra kolonner ved behov
    d[i, paste0("adj_code_", j)]<-v[j]
  }
}
```

Death only event subset and recoding - manual work Originally a “continuation” variable was also included in the export for more information on the event, however, this variable has been excluded from the data set.

```
## If only 1 code, it is in adj_code_1, test if this code is any categorised as "Death"
# subset_death<-d[d$adj_code_1 %in% adj_tbl$code[adj_tbl$soc_term=="Death"] &
#           lengths(ls)==1, ## Redundant test, that only one code was used
#           c("description", "continuation", "event_id")]
## Adds an extra column for adding alternative code manually
# subset_death$add_code<-c("")
# write.csv(subset_death, "subset_death.csv")
```

Hand coded data set imported again

```
head(subset_death_coded<-read_xlsx("subset_death_coded.xlsx")) ## Eight (8) cases, 5 had a new code added
```

```
## # A tibble: 6 x 3
##   description          event_id add_code
##   <chr>                <dbl> <chr>
## 1 301 DSMB                318 410
## 2 302 DSMB                335 <NA>
## 3 307 DSMB                362 104
## 4 312 DSMB2              441 <NA>
## 5 316                    456 <NA>
## 6 316: cerebralt Ã, dem DSMB2 658 007
```

```
## event_id 335 were not recoded, as two events (also event_id 333) are already created for this same d
for (i in 1:nrow(d)){
  for (j in 1:nrow(subset_death_coded)){
    d$adj_code_2[i]<-ifelse(d$event_id[i]==subset_death_coded$event_id[j],
                          subset_death_coded$add_code[j],d$adj_code_2[i])
  }
}
```

```
# subset_801<-d[grepl("801",d$description),c("description","continuation","event_id")]
# write.csv(subset_801,"subset_801.csv")
```

Subset events coded with “801” Every event has been coded with soc_code alternative to 801 or NONE if deemed irrelevant based on other codes at same event.

```
head(alt_801<-read_excel("subset_801_alt.xlsx") %>% na.omit)
```

```
## # A tibble: 6 x 2
##   event_id alt_801
##   <dbl>   <dbl>
## 1     36 10028395
## 2     38 10028395
## 3     50 10028395
## 4     52 10015919
## 5     53 10015919
## 6     63 10028395
```

Converting to new, long data.frame

All events with bleeding (severity) or death are excluded. Death counts will be added later.

```
library(tidyr)
# dput(names(d))
dta<-pivot_longer(select(d,!matches(c("description", "expected", "status", "CLFint")),
                    starts_with("adj_code_"),
                    names_to="adj_index",
                    values_to = "adj_code") %>% # Pivoting to long format
                  na.omit %>% # Omitting NAs, result is complete case only
                  filter(adj_code %in% adj_tbl$code[adj_tbl$soc_term!="Death"&adj_tbl$soc_term!="Bleeding"]))
# Excluding events marked with Death or Bleeding, as these are additional codes not corresponding to soc.
head(dta)
```

```
## # A tibble: 6 x 11
##       X event_id related  rnumb rtreat fatal related_bin serious group adj_index
##   <int>   <int> <chr>    <int> <chr>  <dbl>      <dbl>   <dbl> <chr> <chr>
## 1     1     1 Sandsyn~  529 Place~    0         1         0 Plac~ adj_code~
## 2     2     3 Ikke re~  436 Place~    0         0         1 Plac~ adj_code~
## 3     2     3 Ikke re~  436 Place~    0         0         1 Plac~ adj_code~
## 4     3     4 Ikke re~  253 Aktiv~    0         0         1 Acti~ adj_code~
## 5     4     5 Ikke re~  253 Aktiv~    0         0         0 Acti~ adj_code~
## 6     5     6 Sandsyn~  316 Place~    0         1         0 Plac~ adj_code~
## # ... with 1 more variable: adj_code <chr>
```

Matching adj_code to soc_code

Adding soc_codes and manually coded alternative categories for 801 codes.

```
for (i in 1:nrow(dta)){
  dta$soc[i]<-adj_tbl$meddra[adj_tbl$code==dta$adj_code[i]]
  for (j in 1:nrow(alt_801)){
    dta$soc[i]<-ifelse(dta$event_id[i]==alt_801$event_id[j]&dta$adj_code[i]=="801",
                      alt_801$alt_801[j],dta$soc[i])
  }
}
```

Few were not coded, omitting NAs.

```
dta <- dta %>% na.omit # Only keeping correctly coded cases
```

Adding term and subcat

```
# Subcategory name is the combined soc_term and adj_subcat for additional details in the final report.
for (i in 1:nrow(dta)){
  dta$term[i]<-soc_code$soc_term[soc_code$meddra==dta$soc[i]]
  dta$subcat[i]<-paste0(dta$term[i],": ",adj_tbl$subcat[adj_tbl$code==dta$adj_code[i]])
}
```

Configuring XML

Creating specified data frame

```
df<-data.frame(subjid=dta$rnumb,
               term=dta$subcat,
               soc=as.integer(dta$soc),
               serious=dta$serious,
               related=dta$related_bin,
               fatal=dta$fatal,
               group=dta$group)
head(df)
```

```

##   subjid
## 1    529
## 2    436
## 3    436
## 4    253
## 5    253
## 6    316
##
##                                     term
## 1      Gastrointestinal disorders: Gastro-Intestinal - Other (Specify)
## 2 Respiratory, thoracic and mediastinal disorders: Respiratory - Other (Specify)
## 3      Surgical and medical procedures: Surgery - Other (Specify)
## 4      Surgical and medical procedures: Surgery - Other (Specify)
## 5      Gastrointestinal disorders: Gastro-Intestinal - Other (Specify)
## 6      Nervous system disorders: Neurological - Other (Specify)
##   soc serious related fatal  group
## 1 10017947     0     1     0 Placebo
## 2 10038738     1     0     0 Placebo
## 3 10042613     1     0     0 Placebo
## 4 10042613     1     0     0 Active
## 5 10017947     0     0     0 Active
## 6 10029205     0     1     0 Placebo

```

Handling Deaths

```

# Deaths in named integer vector
# ae_deaths<-table(df$fatal,df$group)[2,] # No deaths included in the sample data set
# These are all the deaths observed within 6 months after randomisation, eg after intention-to-treat
all_deaths<-c("Active"=16,"Placebo"=12)
excess_death<-all_deaths #-ae_deaths

```

Creating safety summary

```

safe_sum<-safety_summary(data=df,
  exposed=c("Active"=319,"Placebo"=323),
  excess_deaths=excess_death)

```

```

simple <- tempfile(fileext = ".xml")
eudract <- tempfile(fileext = ".xml")
simple_safety_xml(safe_sum, simple)

```

```

eudract_convert(input=simple,
  output=eudract)

```