SQL code for verifying results in RDF data cube

mja@statgroup.dk

2016-03-07

Derive results in RDF data cube and compare with results in data cube

Setup

Here all libraries are loaded; this should not be necessary.

```
options(width=200) # long lines
library(xlsx)
```

```
## Loading required package: rJava
```

```
## Loading required package: methods
```

```
## Loading required package: xlsxjars
```

```
library(foreign)
library(rrdf)
```

```
## Loading required package: rrdflibs
```

```
library(rrdfqb)
```

```
## Loading required package: RCurl
```

```
## Loading required package: bitops
```

```
##
## Attaching package: 'RCurl'
```

```
## The following object is masked from 'package:rJava':
##
##     clone
```

```
## Loading required package: rrdfancillary
```

```
library(rrdfqbcrnd0)
```

```
## Loading required package: rrdfcdisc
```

```
## Loading required package: devtools
```

```
##
## Attaching package: 'rrdfqbcrnd0'


## The following object is masked from 'package:rrdfcdisc':
##
##     summarize.rdf.noprint
```

```r
library(rrdfqbcrndex)
library(rrdfqbcrndcheck)
```

Load the data

The SAS dataset is loaded, and factors converted to character.

```r
obsFile<- system.file("extdata/sample-xpt", "adsl.xpt", package="rrdfqbcrndex")
adsl<-read.xport(obsFile)
ii <- sapply(adsl, is.factor)
adsl[ii] <- lapply(adsl[ii], as.character)
```

The conversion to character can be avoided by using `library(SASxport)`, see (../../rrdfqbcrndex/inst/data-raw/create-dm-table-as-csv.Rmd).

Load the RDF data cube

The RDF data cube is loaded.

```r
dataCubeFile<- system.file("extdata/sample-rdf", "DC-DM-sample.ttl", package="rrdfqbcrndex")
store <- new.rdf()  # Initialize
cat("Reading turtle definition from ", dataCubeFile, "\n")
```

```
## Reading turtle definition from  /home/ma/R/x86_64-redhat-linux-gnu-library/3.2/rrdfqbcrndex/extdata/
```

```r
temp<- load.rdf(dataCubeFile, format="TURTLE", appendTo= store)
summarize.rdf(store)
```

```
## [1] "Number of triples: 1450"
```

The mean values for the `WEIGHTBL` variable in RDF data cube is:

```r
rq<-'
prefix crnd-measure: <http://www.example.org/dc/measure#>
prefix code:  <http://www.example.org/dc/code/>
prefix crnd-attribute: <http://www.example.org/dc/attribute#>
prefix ds:    <http://www.example.org/dc/dm/ds/>
prefix qb:    <http://purl.org/linked-data/cube#>
prefix crnd-dimension: <http://www.example.org/dc/dimension#>
select * where {
?obs  a                       qb:Observation ;
    qb:dataSet                ds:dataset-DM ;
```

```
        crnd-dimension:factor        code:factor-WEIGHTBL ;
        crnd-dimension:procedure     code:procedure-mean ;
        crnd-dimension:race          code:race-_ALL_ ;
        crnd-dimension:saffl         code:saffl-Y ;
        crnd-dimension:sex           code:sex-_ALL_ ;
        crnd-dimension:trt01a        ?trt01a ;
    crnd-measure:measure         ?measure .
}
'
knitr::kable(data.frame(sparql.rdf( store, rq)))
```

obs	trt01a	measure
ds:obs38	code:trt01a-Xanomeline_High_Dose	70.0047619047619
ds:obs17	code:trt01a-Placebo	62.7593023255814
ds:obs59	code:trt01a-Xanomeline_Low_Dose	67.2795180722892
The SPARQL	query about can be made more generi	c.
## Generate	SQL statements	
From the RD	F data cube loading in the store, t	he function GetSQLFromCube generates the SQL statements for rep
Only the fi	rst two select statements are shown	.

```
stmtSQL<- GetSQLFromCube( store, srcDsName="adsl" )
cat(paste(unlist(strsplit(stmtSQL$summStatSQL,split="\n"))[1:3],collapse="\n"),"\n")
```

```
## SELECT a.TRT01A, '_ALL_' as RACE, '_ALL_' as SEX, a.SAFFL, 'min' as procedure, 'AGE' as factor, '_NU
## UNION
## SELECT a.TRT01A, '_ALL_' as RACE, a.SEX, a.SAFFL, 'count' as procedure, 'quantity' as factor, '_ALL_
```

Derive the descriptive statistics

```
library(sqldf)
```

```
## Loading required package: gsubfn
```

```
## Loading required package: proto
```

```
## Loading required package: RSQLite
```

```
## Loading required package: DBI
```

```
adsl.summ.stat.res<- sqldf( stmtSQL$summStatSQL)
```

```
## Loading required package: tcltk
```

```
names(adsl.summ.stat.res)<- tolower(gsub("(a|b)\\.","", names(adsl.summ.stat.res)))
knitr::kable(adsl.summ.stat.res)
```

trt01a	race	sex	saffl	procedure	factor	denom
Placebo	AMERICAN INDIAN OR ALASKA NATIVE	*ALL*	Y	percent	proportion	RACE
Placebo	BLACK OR AFRICAN AMERICAN	*ALL*	Y	count	quantity	*ALL*
Placebo	BLACK OR AFRICAN AMERICAN	*ALL*	Y	percent	proportion	RACE
Placebo	WHITE	*ALL*	Y	count	quantity	*ALL*
Placebo	WHITE	*ALL*	Y	percent	proportion	RACE
Placebo	*ALL*	F	Y	count	quantity	*ALL*
Placebo	*ALL*	F	Y	percent	proportion	SEX
Placebo	*ALL*	M	Y	count	quantity	*ALL*
Placebo	*ALL*	M	Y	percent	proportion	SEX
Placebo	*ALL*	*ALL*	Y	count	quantity	*ALL*
Placebo	*ALL*	*ALL*	Y	max	AGE	*NULL*
Placebo	*ALL*	*ALL*	Y	max	WEIGHTBL	*NULL*
Placebo	*ALL*	*ALL*	Y	mean	AGE	*NULL*
Placebo	*ALL*	*ALL*	Y	mean	WEIGHTBL	*NULL*
Placebo	*ALL*	*ALL*	Y	median	AGE	*NULL*
Placebo	*ALL*	*ALL*	Y	median	WEIGHTBL	*NULL*
Placebo	*ALL*	*ALL*	Y	min	AGE	*NULL*
Placebo	*ALL*	*ALL*	Y	min	WEIGHTBL	*NULL*
Placebo	*ALL*	*ALL*	Y	stdev	AGE	*NULL*
Placebo	*ALL*	*ALL*	Y	stdev	WEIGHTBL	*NULL*
Xanomeline High Dose	AMERICAN INDIAN OR ALASKA NATIVE	*ALL*	Y	count	quantity	*ALL*
Xanomeline High Dose	AMERICAN INDIAN OR ALASKA NATIVE	*ALL*	Y	percent	proportion	RACE
Xanomeline High Dose	BLACK OR AFRICAN AMERICAN	*ALL*	Y	count	quantity	*ALL*
Xanomeline High Dose	BLACK OR AFRICAN AMERICAN	*ALL*	Y	percent	proportion	RACE
Xanomeline High Dose	WHITE	*ALL*	Y	count	quantity	*ALL*
Xanomeline High Dose	WHITE	*ALL*	Y	percent	proportion	RACE
Xanomeline High Dose	*ALL*	F	Y	count	quantity	*ALL*
Xanomeline High Dose	*ALL*	F	Y	percent	proportion	SEX
Xanomeline High Dose	*ALL*	M	Y	count	quantity	*ALL*
Xanomeline High Dose	*ALL*	M	Y	percent	proportion	SEX
Xanomeline High Dose	*ALL*	*ALL*	Y	count	quantity	*ALL*
Xanomeline High Dose	*ALL*	*ALL*	Y	max	AGE	*NULL*
Xanomeline High Dose	*ALL*	*ALL*	Y	max	WEIGHTBL	*NULL*
Xanomeline High Dose	*ALL*	*ALL*	Y	mean	AGE	*NULL*
Xanomeline High Dose	*ALL*	*ALL*	Y	mean	WEIGHTBL	*NULL*
Xanomeline High Dose	*ALL*	*ALL*	Y	median	AGE	*NULL*
Xanomeline High Dose	*ALL*	*ALL*	Y	median	WEIGHTBL	*NULL*
Xanomeline High Dose	*ALL*	*ALL*	Y	min	AGE	*NULL*
Xanomeline High Dose	*ALL*	*ALL*	Y	min	WEIGHTBL	*NULL*
Xanomeline High Dose	*ALL*	*ALL*	Y	stdev	AGE	*NULL*
Xanomeline High Dose	*ALL*	*ALL*	Y	stdev	WEIGHTBL	*NULL*
Xanomeline Low Dose	AMERICAN INDIAN OR ALASKA NATIVE	*ALL*	Y	percent	proportion	RACE
Xanomeline Low Dose	BLACK OR AFRICAN AMERICAN	*ALL*	Y	count	quantity	*ALL*
Xanomeline Low Dose	BLACK OR AFRICAN AMERICAN	*ALL*	Y	percent	proportion	RACE
Xanomeline Low Dose	WHITE	*ALL*	Y	count	quantity	*ALL*
Xanomeline Low Dose	WHITE	*ALL*	Y	percent	proportion	RACE
Xanomeline Low Dose	*ALL*	F	Y	count	quantity	*ALL*
Xanomeline Low Dose	*ALL*	F	Y	percent	proportion	SEX
Xanomeline Low Dose	*ALL*	M	Y	count	quantity	*ALL*
Xanomeline Low Dose	*ALL*	M	Y	percent	proportion	SEX
Xanomeline Low Dose	*ALL*	*ALL*	Y	count	quantity	*ALL*
Xanomeline Low Dose	*ALL*	*ALL*	Y	max	AGE	*NULL*

trt01a	race	sex	saffl	procedure	factor	denom
Xanomeline Low Dose	*ALL*	*ALL*	Y	max	WEIGHTBL	*NULL*
Xanomeline Low Dose	*ALL*	*ALL*	Y	mean	AGE	*NULL*
Xanomeline Low Dose	*ALL*	*ALL*	Y	mean	WEIGHTBL	*NULL*
Xanomeline Low Dose	*ALL*	*ALL*	Y	median	AGE	*NULL*
Xanomeline Low Dose	*ALL*	*ALL*	Y	median	WEIGHTBL	*NULL*
Xanomeline Low Dose	*ALL*	*ALL*	Y	min	AGE	*NULL*
Xanomeline Low Dose	*ALL*	*ALL*	Y	min	WEIGHTBL	*NULL*
Xanomeline Low Dose	*ALL*	*ALL*	Y	stdev	AGE	*NULL*
Xanomeline Low Dose	*ALL*	*ALL*	Y	stdev	WEIGHTBL	*NULL*
ALL	AMERICAN INDIAN OR ALASKA NATIVE	*ALL*	Y	count	quantity	*ALL*
ALL	BLACK OR AFRICAN AMERICAN	*ALL*	Y	count	quantity	*ALL*
ALL	WHITE	*ALL*	Y	count	quantity	*ALL*
ALL	*ALL*	F	Y	count	quantity	*ALL*
ALL	*ALL*	M	Y	count	quantity	*ALL*

Quick check

Sometimes it is usefull to see that there is an actual derivation going on. The calculated mean values for the `WEIGHTBL` variable is

```
knitr::kable(adsl.summ.stat.res[adsl.summ.stat.res$factor=="WEIGHTBL" & adsl.summ.stat.res$procedure=="
```

	trt01a	race	sex	saffl	procedure	factor	denominator	unit	measure
14	Placebo	*ALL*	*ALL*	Y	mean	WEIGHTBL	*NULL*	KG	62.75930
35	Xanomeline High Dose	*ALL*	*ALL*	Y	mean	WEIGHTBL	*NULL*	KG	70.00476
55	Xanomeline Low Dose	*ALL*	*ALL*	Y	mean	WEIGHTBL	*NULL*	KG	67.27952

If `WEIGHTBL` is multiplied by 100, then mean should also be multiplied with 100:

```
adsl$WEIGHTBL<- adsl$WEIGHTBL*100
adsl.summ.stat.mod.res<- sqldf( stmtSQL$summStatSQL)
knitr::kable(adsl.summ.stat.mod.res[adsl.summ.stat.mod.res$factor=="WEIGHTBL" & adsl.summ.stat.mod.res$p
```

	a.TRT01A	RACE	SEX	a.SAFFL	procedure	factor	denominator	unit	measure
14	Placebo	*ALL*	*ALL*	Y	mean	WEIGHTBL	*NULL*	KG	6275.930
35	Xanomeline High Dose	*ALL*	*ALL*	Y	mean	WEIGHTBL	*NULL*	KG	7000.476
55	Xanomeline Low Dose	*ALL*	*ALL*	Y	mean	WEIGHTBL	*NULL*	KG	6727.952