

Using ARQ to make SPARQL queries

mja@statgroup.dk

2016-04-26

Contents

SPARQL scripts for the demographics cube (DC-DEMO-sample.ttl)	1
Get dimensions	1
Get attributes	19
Get observations	36
Get definition for all descriptive statistics - class <code>code:procedure</code>	56
Get the codelists	74
Get the codelist definition for all descriptive statistics - class <code>code:procedure</code>	92
Get definition for all variables used for descriptive statistics - class <code>code:factor</code>	111
Get the codelist definition for all variables used for descriptive statistics - class <code>code:factor</code>	128
Get definition for descriptive statistics median	147
Get information for selection of data	164
Get information on the underlying data in D2RQ format	182
Get underlying data for one cube observation	184
 How to run this .Rmd file	 201

SPARQL scripts for the demographics cube (DC-DEMO-sample.ttl)

The examples below uses `arq` from Apache Jena (<http://jena.apache.org>). To install `arq` - download and unpack the latest version of `apache-jena` from (<http://jena.apache.org/download/index.cgi>). Then you need some way of invoking `arq`; I use a not-so-clever-approach: `cd ~/bin; ln -s /opt/apache-jena-2.13.0/bin/arq`.

Given a SPARQL query and RDF data, `arq` returns the result of the query. So this is the command line way of making a SPARQL query.

The use of `arq` is described many places, see for example (<http://www.learningsparql.com/>).

All `arq` commands below are to be run in the directory with the sample files, which is `inst/extdata/sample-rdf` directory or `extdata/sample-rdf` depending on the whether the development version or the installed version of the package is used.

The `cd` below in each code block is included because I could not find a quick way to get the code chunk executed in that directory. `knitr` is flexible enough to do it, I have not yet found the right way to do it. So, ignore the repeated `cd ..`

Get dimensions

```
cd ../extdata/sample-rdf
arq --data DC-DEMO-sample.ttl --query DEMOdDimensions.rq
```

[illegible]

[illegible]

[illegible]

5

[illegible]

7

[illegible]

9

[illegible]

[illegible]

[illegible]

[illegible]

14

[illegible]

[illegible]

[illegible]

[illegible]

```

## 12:00:49 WARN riot :: [line: 3193, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:49 WARN riot :: [line: 3194, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:49 WARN riot :: [line: 3195, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:49 WARN riot :: [line: 3211, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:49 WARN riot :: [line: 3212, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:49 WARN riot :: [line: 3213, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:49 WARN riot :: [line: 3214, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:49 WARN riot :: [line: 3215, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:49 WARN riot :: [line: 3216, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:49 WARN riot :: [line: 3217, col: 37] Bad IRI: <http://www.example.org/dc/code/
## -----
## | p |
## =====
## | crnd-dimension:ethnic |
## | crnd-dimension:race |
## | crnd-dimension:procedure |
## | crnd-dimension:agegr1 |
## | crnd-dimension:factor |
## | crnd-dimension:trt01a |
## | crnd-dimension:sex |
## -----

```

Get attributes

```

cd ../extdata/sample-rdf
arq --data DC-DEMO-sample.ttl --query DEMOattributes.rq

```

```

## 12:00:50 WARN riot :: [line: 47, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 48, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 49, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 50, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 51, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 52, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 53, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 66, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 67, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 68, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 69, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 70, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 71, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 72, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 94, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 95, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 96, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 97, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 98, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 99, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 100, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 113, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 114, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 115, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:50 WARN riot :: [line: 116, col: 37] Bad IRI: <http://www.example.org/dc/code/

```

20

[illegible]

22

23

24

[illegible]

[illegible]

27

[illegible]

29

[illegible]

[illegible]

[illegible]

33

[illegible]

35

```

## 12:00:50 WARN riot :: [line: 3119, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3120, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3121, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3122, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3123, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3124, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3125, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3138, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3139, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3140, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3141, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3142, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3143, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3144, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3170, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3171, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3172, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3173, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3174, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3175, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3176, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3189, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3190, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3191, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3192, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3193, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3194, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3195, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3211, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3212, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3213, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3214, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3215, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3216, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:50 WARN riot :: [line: 3217, col: 37] Bad IRI: <http://www.example.org/dc/cod
## -----
## | p |
## =====
## | crnd-attribute:cellpartno |
## | crnd-attribute:measurefmt |
## | crnd-attribute:colno |
## | crnd-attribute:denominator |
## | crnd-attribute:unit |
## | crnd-attribute:rowno |
## -----

```

Get observations

The SPARQL script shows for each observation the dimension, attributes and measures in a row. Note: in the HTML version the output below can be scrolled using the left and right arrow.

```

cd ../extdata/sample-rdf
arq --data DC-DEMO-sample.ttl --query DEMOobservations.rq

```

[illegible]

[illegible]

[illegible]

40

41

42

[illegible]

[illegible]

[illegible]

[illegible]

47

[illegible]

[illegible]

[illegible]

[illegible]

52

53

[illegible]

[illegible]

[illegible]

58

[illegible]

[illegible]

[illegible]

62

[illegible]

[illegible]

65

66

67

[illegible]

69

[illegible]

[illegible]

[illegible]

[illegible]

```
## | code:procedure-mean      | "mean"^^xsd:string | "function (x) {      mean(x, na.rm = TRUE) }"
## | code:procedure-min      | "min"^^xsd:string  | "function (x) {      min(x, na.rm = TRUE) }"
## | code:procedure-percent  | "percent"^^xsd:string | "function (x) {      -1 }"
## | code:procedure-count    | "count"^^xsd:string | "function (x) {      length(x) }"
## | code:procedure-q3       | "q3"^^xsd:string   | "function (x) {      quantile(x, probs = c(0.75),
## | code:procedure-n        | "n"^^xsd:string     | "function (x) {      length(x[!is.na(x)]) }"
## | code:procedure-median   | "median"^^xsd:string | "function (x) {      median(x, na.rm = TRUE) }"
## | code:procedure-q1       | "q1"^^xsd:string    | "function (x) {      quantile(x, probs = c(0.25),
## -----
```

Get the codelists

The SPARQL script shows the codelist.

```
cd ../extdata/sample-rdf
arq --data DC-DEMO-sample.ttl --query DEMOcodelist.rq
```

```
## 12:00:54 WARN riot :: [line: 47, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 48, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 49, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 50, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 51, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 52, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 53, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 66, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 67, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 68, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 69, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 70, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 71, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 72, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 94, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 95, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 96, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 97, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 98, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 99, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 100, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 113, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 114, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 115, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 116, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 117, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 118, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 119, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 132, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 133, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 134, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 135, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 136, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 137, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 138, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:54 WARN riot :: [line: 176, col: 37] Bad IRI: <http://www.example.org/dc/code/
```

75

76

77

78

79

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

85

[illegible]

87

88

[illegible]

[illegible]


```

## | ds:dsd-DEMO          | crnd-dimension:race      | "Codelist scheme: race"@en | code:race-BL
## | ds:dsd-DEMO          | crnd-dimension:race      | "Codelist scheme: race"@en | code:race-NA
## | ds:dsd-DEMO          | crnd-dimension:race      | "Codelist scheme: race"@en | code:race-WI
## | ds:dsd-DEMO          | crnd-dimension:race      | "Codelist scheme: race"@en | code:race-_L
## | ds:dsd-DEMO          | crnd-dimension:race      | "Codelist scheme: race"@en | code:race-_L
## | ds:dsd-DEMO          | crnd-dimension:sex       | "Codelist scheme: sex"@en   | code:sex-F
## | ds:dsd-DEMO          | crnd-dimension:sex       | "Codelist scheme: sex"@en   | code:sex-M
## | ds:dsd-DEMO          | crnd-dimension:sex       | "Codelist scheme: sex"@en   | code:sex-U
## | ds:dsd-DEMO          | crnd-dimension:sex       | "Codelist scheme: sex"@en   | code:sex-UN
## | ds:dsd-DEMO          | crnd-dimension:sex       | "Codelist scheme: sex"@en   | code:sex-_A
## | ds:dsd-DEMO          | crnd-dimension:sex       | "Codelist scheme: sex"@en   | code:sex-_N
## | ds:dsd-DEMO          | crnd-dimension:trt01a    | "Codelist scheme: trt01a"@en | code:trt01a-
## | ds:dsd-DEMO          | crnd-dimension:trt01a    | "Codelist scheme: trt01a"@en | code:trt01a-
## | ds:dsd-DEMO          | crnd-dimension:trt01a    | "Codelist scheme: trt01a"@en | code:trt01a-
## | ds:dsd-DEMO          | crnd-dimension:trt01a    | "Codelist scheme: trt01a"@en | code:trt01a-
## | ds:dsd-DEMO          | crnd-dimension:trt01a    | "Codelist scheme: trt01a"@en | code:trt01a-
## -----

```

Get the codelist definition for all descriptive statistics - class code:procedure

The SPARQL script shows how the R function definition for the descriptive statistics is stored in the cube.

```

cd ../extdata/sample-rdf
arq --data DC-DEMO-sample.ttl --query DEMOprocedure-codelist.rq

```

```

## 12:00:55 WARN riot :: [line: 47, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 48, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 49, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 50, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 51, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 52, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 53, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 66, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 67, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 68, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 69, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 70, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 71, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 72, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 94, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 95, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 96, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 97, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 98, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 99, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 100, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 113, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 114, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 115, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 116, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 117, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 118, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:55 WARN riot :: [line: 119, col: 37] Bad IRI: <http://www.example.org/dc/code/

```

93

94

95

96

97

[illegible]

99

100

101

[illegible]

103

[illegible]

105

106

107

108

```

## 12:00:55 WARN riot :: [line: 3122, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3123, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3124, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3125, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3138, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3139, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3140, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3141, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3142, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3143, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3144, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3170, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3171, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3172, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3173, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3174, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3175, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3176, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3189, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3190, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3191, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3192, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3193, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3194, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3195, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3211, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3212, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3213, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3214, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3215, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3216, col: 37] Bad IRI: <http://www.example.org/dc/code
## 12:00:55 WARN riot :: [line: 3217, col: 37] Bad IRI: <http://www.example.org/dc/code
## -----
## | s | p | o |
## =====
## | code:procedure-min | skos:topConceptOf | code:procedure
## | code:procedure-min | skos:prefLabel | "min"^^xsd:string
## | code:procedure-min | skos:inScheme | code:procedure
## | code:procedure-min | rrdqbcrnd0:RdescStatDefFun | "function (x) { min(x, na.rm = TRUE)
## | code:procedure-min | rrdqbcrnd0:R-selectionvalue | "min"^^xsd:string
## | code:procedure-min | rrdqbcrnd0:R-selectionoperator | "=="^^xsd:string
## | code:procedure-min | rdfs:comment | "Descriptive statistics min"@en
## | code:procedure-min | rdf:type | skos:Concept
## | code:procedure-min | rdf:type | code:Procedure
## | code:procedure-std | skos:topConceptOf | code:procedure
## | code:procedure-std | skos:prefLabel | "std"^^xsd:string
## | code:procedure-std | skos:inScheme | code:procedure
## | code:procedure-std | rrdqbcrnd0:RdescStatDefFun | "function (x) { sd(x, na.rm = TRUE)
## | code:procedure-std | rrdqbcrnd0:R-selectionvalue | "std"^^xsd:string
## | code:procedure-std | rrdqbcrnd0:R-selectionoperator | "=="^^xsd:string
## | code:procedure-std | rdfs:comment | "Descriptive statistics std"@en
## | code:procedure-std | rdf:type | skos:Concept
## | code:procedure-std | rdf:type | code:Procedure
## | code:procedure-q3 | skos:topConceptOf | code:procedure

```

##		code:procedure-q3		skos:prefLabel		"q3"^^xsd:string
##		code:procedure-q3		skos:inScheme		code:procedure
##		code:procedure-q3		rrdfqbcrnd0:RdescStatDefFun		"function (x) { quantile(x, probs =
##		code:procedure-q3		rrdfqbcrnd0:R-selectionvalue		"q3"^^xsd:string
##		code:procedure-q3		rrdfqbcrnd0:R-selectionoperator		"=="^^xsd:string
##		code:procedure-q3		rdfs:comment		"Descriptive statistics q3"@en
##		code:procedure-q3		rdf:type		skos:Concept
##		code:procedure-q3		rdf:type		code:Procedure
##		code:procedure-median		skos:topConceptOf		code:procedure
##		code:procedure-median		skos:prefLabel		"median"^^xsd:string
##		code:procedure-median		skos:inScheme		code:procedure
##		code:procedure-median		rrdfqbcrnd0:RdescStatDefFun		"function (x) { median(x, na.rm =
##		code:procedure-median		rrdfqbcrnd0:R-selectionvalue		"median"^^xsd:string
##		code:procedure-median		rrdfqbcrnd0:R-selectionoperator		"=="^^xsd:string
##		code:procedure-median		rdfs:comment		"Descriptive statistics median"@en
##		code:procedure-median		rdf:type		skos:Concept
##		code:procedure-median		rdf:type		code:Procedure
##		code:procedure-count		skos:topConceptOf		code:procedure
##		code:procedure-count		skos:prefLabel		"count"^^xsd:string
##		code:procedure-count		skos:inScheme		code:procedure
##		code:procedure-count		rrdfqbcrnd0:RdescStatDefFun		"function (x) { length(x) }"
##		code:procedure-count		rrdfqbcrnd0:R-selectionvalue		"count"^^xsd:string
##		code:procedure-count		rrdfqbcrnd0:R-selectionoperator		"=="^^xsd:string
##		code:procedure-count		rdfs:comment		"Descriptive statistics count"@en
##		code:procedure-count		rdf:type		skos:Concept
##		code:procedure-count		rdf:type		code:Procedure
##		code:procedure-max		skos:topConceptOf		code:procedure
##		code:procedure-max		skos:prefLabel		"max"^^xsd:string
##		code:procedure-max		skos:inScheme		code:procedure
##		code:procedure-max		rrdfqbcrnd0:RdescStatDefFun		"function (x) { max(x, na.rm = TRUE
##		code:procedure-max		rrdfqbcrnd0:R-selectionvalue		"max"^^xsd:string
##		code:procedure-max		rrdfqbcrnd0:R-selectionoperator		"=="^^xsd:string
##		code:procedure-max		rdfs:comment		"Descriptive statistics max"@en
##		code:procedure-max		rdf:type		skos:Concept
##		code:procedure-max		rdf:type		code:Procedure
##		code:procedure-mean		skos:topConceptOf		code:procedure
##		code:procedure-mean		skos:prefLabel		"mean"^^xsd:string
##		code:procedure-mean		skos:inScheme		code:procedure
##		code:procedure-mean		rrdfqbcrnd0:RdescStatDefFun		"function (x) { mean(x, na.rm = TRUE
##		code:procedure-mean		rrdfqbcrnd0:R-selectionvalue		"mean"^^xsd:string
##		code:procedure-mean		rrdfqbcrnd0:R-selectionoperator		"=="^^xsd:string
##		code:procedure-mean		rdfs:comment		"Descriptive statistics mean"@en
##		code:procedure-mean		rdf:type		skos:Concept
##		code:procedure-mean		rdf:type		code:Procedure
##		code:procedure-q1		skos:topConceptOf		code:procedure
##		code:procedure-q1		skos:prefLabel		"q1"^^xsd:string
##		code:procedure-q1		skos:inScheme		code:procedure
##		code:procedure-q1		rrdfqbcrnd0:RdescStatDefFun		"function (x) { quantile(x, probs =
##		code:procedure-q1		rrdfqbcrnd0:R-selectionvalue		"q1"^^xsd:string
##		code:procedure-q1		rrdfqbcrnd0:R-selectionoperator		"=="^^xsd:string
##		code:procedure-q1		rdfs:comment		"Descriptive statistics q1"@en
##		code:procedure-q1		rdf:type		skos:Concept
##		code:procedure-q1		rdf:type		code:Procedure
##		code:procedure-n		skos:topConceptOf		code:procedure

```

## | code:procedure-n      | skos:prefLabel      | "n"^^xsd:string
## | code:procedure-n      | skos:inScheme        | code:procedure
## | code:procedure-n      | rrdqbcrnd0:RdescStatDefFun | "function (x) {      length(x[!is.na(x)]
## | code:procedure-n      | rrdqbcrnd0:R-selectionvalue | "n"^^xsd:string
## | code:procedure-n      | rrdqbcrnd0:R-selectionoperator | "=="^^xsd:string
## | code:procedure-n      | rdfs:comment          | "Descriptive statistics n"@en
## | code:procedure-n      | rdf:type              | skos:Concept
## | code:procedure-n      | rdf:type              | code:Procedure
## | code:procedure-percent | skos:topConceptOf     | code:procedure
## | code:procedure-percent | skos:prefLabel        | "percent"^^xsd:string
## | code:procedure-percent | skos:inScheme          | code:procedure
## | code:procedure-percent | rrdqbcrnd0:RdescStatDefFun | "function (x) {      -1 }"
## | code:procedure-percent | rrdqbcrnd0:R-selectionvalue | "percent"^^xsd:string
## | code:procedure-percent | rrdqbcrnd0:R-selectionoperator | "=="^^xsd:string
## | code:procedure-percent | rdfs:comment           | "Descriptive statistics percent"@en
## | code:procedure-percent | rdf:type              | skos:Concept
## | code:procedure-percent | rdf:type              | code:Procedure
## -----

```

Get definition for all variables used for descriptive statistics - class `code:factor`

The SPARQL script shows how the R function definition for the descriptive statistics is stored in the cube.

```

cd ../extdata/sample-rdf
arq --data DC-DEMO-sample.ttl --query DEMOfactor.rq

```

```

## 12:00:56 WARN riot :: [line: 47, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 48, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 49, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 50, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 51, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 52, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 53, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 66, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 67, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 68, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 69, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 70, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 71, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 72, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 94, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 95, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 96, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 97, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 98, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 99, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 100, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 113, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 114, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 115, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 116, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 117, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:56 WARN riot :: [line: 118, col: 37] Bad IRI: <http://www.example.org/dc/code/

```

[illegible]

[illegible]

[illegible]

115

116

117

118

119

120

[illegible]

[illegible]

[illegible]

124

125

126

127

```

## 12:00:56 WARN riot :: [line: 3121, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3122, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3123, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3124, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3125, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3138, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3139, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3140, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3141, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3142, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3143, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3144, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3170, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3171, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3172, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3173, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3174, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3175, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3176, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3189, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3190, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3191, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3192, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3193, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3194, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3195, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3211, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3212, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3213, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3214, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3215, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3216, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:56 WARN riot :: [line: 3217, col: 37] Bad IRI: <http://www.example.org/dc/cod
## -----
## | factor | prefLabel | DataSetRefD2RQ | Rselectionvalue
## =====
## | code:factor-_ALL_ | "_ALL_"^^xsd:string | |
## | code:factor-proportion | "proportion"^^xsd:string | | "proportion"^^xsd:st
## | code:factor-weightbl | "weightbl"^^xsd:string | rrdfqbc rnd0:ADSL_WEIGHTBL | "weightbl"^^xsd:st
## | code:factor-_NONMISS_ | "_NONMISS_"^^xsd:string | |
## | code:factor-quantity | "quantity"^^xsd:string | | "quantity"^^xsd:st
## | code:factor-age | "age"^^xsd:string | rrdfqbc rnd0:ADSL_AGE | "age"^^xsd:string
## -----

```

Get the codelist definition for all variables used for descriptive statistics - class `code:factor`

The SPARQL script shows how the R function definition for the descriptive statistics is stored in the cube.

```

cd ../extdata/sample-rdf
arq --data DC-DEMO-sample.ttl --query DEMOfactor-codelist.rq

```

```

## 12:00:57 WARN riot :: [line: 47, col: 37] Bad IRI: <http://www.example.org/dc/code/

```


[illegible]

130

131

132

133

[illegible]

135

136

137

[illegible]

139

[illegible]

141

[illegible]

[illegible]

[illegible]

[illegible]

```

## 12:00:57 WARN riot :: [line: 3213, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:57 WARN riot :: [line: 3214, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:57 WARN riot :: [line: 3215, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:57 WARN riot :: [line: 3216, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:00:57 WARN riot :: [line: 3217, col: 37] Bad IRI: <http://www.example.org/dc/cod
## -----
## | s | p | o |
## =====
## | code:factor-weightbl | skos:topConceptOf | code:factor |
## | code:factor-weightbl | skos:prefLabel | "weightbl"^^xsd:string |
## | code:factor-weightbl | skos:inScheme | code:factor |
## | code:factor-weightbl | rrdqbcrnd0:R-selectionvalue | "weightbl"^^xsd:string |
## | code:factor-weightbl | rrdqbcrnd0:R-selectionoperator | "=="^^xsd:string |
## | code:factor-weightbl | rrdqbcrnd0:DataSetRefD2RQ | rrdqbcrnd0:ADSL_WEIGHTBL |
## | code:factor-weightbl | rdfs:comment | "Coded values from data source. No recor |
## | code:factor-weightbl | rdf:type | skos:Concept |
## | code:factor-weightbl | rdf:type | code:Factor |
## | code:factor-age | skos:topConceptOf | code:factor |
## | code:factor-age | skos:prefLabel | "age"^^xsd:string |
## | code:factor-age | skos:inScheme | code:factor |
## | code:factor-age | rrdqbcrnd0:R-selectionvalue | "age"^^xsd:string |
## | code:factor-age | rrdqbcrnd0:R-selectionoperator | "=="^^xsd:string |
## | code:factor-age | rrdqbcrnd0:DataSetRefD2RQ | rrdqbcrnd0:ADSL_AGE |
## | code:factor-age | rdfs:comment | "Coded values from data source. No recor |
## | code:factor-age | rdf:type | skos:Concept |
## | code:factor-age | rdf:type | code:Factor |
## | code:factor-_ALL_ | skos:topConceptOf | code:factor |
## | code:factor-_ALL_ | skos:prefLabel | "_ALL_"^^xsd:string |
## | code:factor-_ALL_ | skos:inScheme | code:factor |
## | code:factor-_ALL_ | rdfs:comment | "NON-CDISC: Represents all codelist cat |
## | code:factor-_ALL_ | rdf:type | skos:Concept |
## | code:factor-_ALL_ | rdf:type | code:Factor |
## | code:factor-quantity | skos:topConceptOf | code:factor |
## | code:factor-quantity | skos:prefLabel | "quantity"^^xsd:string |
## | code:factor-quantity | skos:inScheme | code:factor |
## | code:factor-quantity | rrdqbcrnd0:R-selectionvalue | "quantity"^^xsd:string |
## | code:factor-quantity | rrdqbcrnd0:R-selectionoperator | "=="^^xsd:string |
## | code:factor-quantity | rdfs:comment | "Coded values from data source. No recor |
## | code:factor-quantity | rdf:type | skos:Concept |
## | code:factor-quantity | rdf:type | code:Factor |
## | code:factor-_NONMISS_ | skos:topConceptOf | code:factor |
## | code:factor-_NONMISS_ | skos:prefLabel | "_NONMISS_"^^xsd:string |
## | code:factor-_NONMISS_ | skos:inScheme | code:factor |
## | code:factor-_NONMISS_ | rrdqbcrnd0:R-selectionfunction | "is.na"^^xsd:string |
## | code:factor-_NONMISS_ | rdfs:comment | "NON-CDISC: Represents the non-missing |
## | code:factor-_NONMISS_ | rdf:type | skos:Concept |
## | code:factor-_NONMISS_ | rdf:type | code:Factor |
## | code:factor-proportion | skos:topConceptOf | code:factor |
## | code:factor-proportion | skos:prefLabel | "proportion"^^xsd:string |
## | code:factor-proportion | skos:inScheme | code:factor |
## | code:factor-proportion | rrdqbcrnd0:R-selectionvalue | "proportion"^^xsd:string |
## | code:factor-proportion | rrdqbcrnd0:R-selectionoperator | "=="^^xsd:string |
## | code:factor-proportion | rdfs:comment | "Coded values from data source. No recor |
## | code:factor-proportion | rdf:type | skos:Concept |

```

```
## | code:factor-proportion | rdf:type | code:Factor
## -----
```

Get definition for descriptive statistics median

The SPARQL script shows how the function definition for the descriptive statistics is stored in the cube.

```
cd ../extdata/sample-rdf
arq --data DC-DEMO-sample.ttl --query DEMOprocedure-median.rq
```

```
## 12:00:58 WARN riot :: [line: 47, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 48, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 49, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 50, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 51, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 52, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 53, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 66, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 67, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 68, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 69, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 70, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 71, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 72, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 94, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 95, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 96, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 97, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 98, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 99, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 100, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 113, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 114, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 115, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 116, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 117, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 118, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 119, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 132, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 133, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 134, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 135, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 136, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 137, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 138, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 176, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 177, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 178, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 179, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 180, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 181, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 182, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 195, col: 37] Bad IRI: <http://www.example.org/dc/code/
```

148

149

150

[illegible]

[illegible][illegible]

[illegible]

154

155

156

157

158

[illegible]

[illegible]

[illegible]

[illegible]

163

```

## 12:00:58 WARN riot :: [line: 3174, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 3175, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 3176, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 3189, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 3190, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 3191, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 3192, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 3193, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 3194, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 3195, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 3211, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 3212, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 3213, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 3214, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 3215, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 3216, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:58 WARN riot :: [line: 3217, col: 37] Bad IRI: <http://www.example.org/dc/code/
## -----
## | s | p | o |
## =====
## | code:procedure-median | skos:topConceptOf | code:procedure |
## | code:procedure-median | skos:prefLabel | "median"^^xsd:string |
## | code:procedure-median | skos:inScheme | code:procedure |
## | code:procedure-median | rrdfqbcrrnd0:RdescStatDefFun | "function (x) { median(x, na.rm = T |
## | code:procedure-median | rrdfqbcrrnd0:R-selectionvalue | "median"^^xsd:string |
## | code:procedure-median | rrdfqbcrrnd0:R-selectionoperator | "=="^^xsd:string |
## | code:procedure-median | rdfs:comment | "Descriptive statistics median"@en |
## | code:procedure-median | rdf:type | skos:Concept |
## | code:procedure-median | rdf:type | code:Procedure |
## -----

```

Get information for selection of data

The SPARQL script shows how the information for selecting data for derivation of univariate statistics is present in the cube.

```

cd ../extdata/sample-rdf
arq --data DC-DEMO-sample.ttl --query DEMOobservations-R-selection.rq

```

```

## 12:00:59 WARN riot :: [line: 47, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:59 WARN riot :: [line: 48, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:59 WARN riot :: [line: 49, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:59 WARN riot :: [line: 50, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:59 WARN riot :: [line: 51, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:59 WARN riot :: [line: 52, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:59 WARN riot :: [line: 53, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:59 WARN riot :: [line: 66, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:59 WARN riot :: [line: 67, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:59 WARN riot :: [line: 68, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:59 WARN riot :: [line: 69, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:59 WARN riot :: [line: 70, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:59 WARN riot :: [line: 71, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:00:59 WARN riot :: [line: 72, col: 37] Bad IRI: <http://www.example.org/dc/code/

```

[illegible]

166

167

168

169

[illegible]

[illegible]

[illegible]

173

174

175

176

177

178

179

180

```
## -----
## | obs | rrdfqbcrrndORcolumnname | Rselectionoperator | Rselectionvalue |
## =====
## -----
```

Get information on the underlying data in D2RQ format

The following two SPARQL scripts are from taken from `create-ADSL-ttl.Rmd`.

First get the mapping information.

```
cd ../extdata/sample-rdf
arq --data adsl-map.ttl --query ADSL-mapping.rq
```

```
## -----
## | mapColumn                | d2rqcolumn      | d2rqdatatype |
## =====
## | <adsl-map.ttl#ADSL_COMP24FL> | "ADSL.COMP24FL" |              |
## | <adsl-map.ttl#ADSL_DCREASCD> | "ADSL.DCREASCD" |              |
## | <adsl-map.ttl#ADSL_DTHFL>    | "ADSL.DTHFL"    |              |
## | <adsl-map.ttl#ADSL_TRTSDT>   | "ADSL.TRTSDT"   | xsd:double   |
## | <adsl-map.ttl#ADSL_RFENDT>   | "ADSL.RFENDT"   | xsd:double   |
## | <adsl-map.ttl#ADSL_DCDECOD>  | "ADSL.DCDECOD"  |              |
## | <adsl-map.ttl#ADSL_RFENDTC>  | "ADSL.RFENDTC"  |              |
## | <adsl-map.ttl#ADSL_CUMDOSE>  | "ADSL.CUMDOSE"  | xsd:double   |
## | <adsl-map.ttl#ADSL_TRT01A>   | "ADSL.TRT01A"   |              |
## | <adsl-map.ttl#ADSL_AGEGR1>   | "ADSL.AGEGR1"   |              |
## | <adsl-map.ttl#ADSL_USUBJID>  | "ADSL.USUBJID"  |              |
## | <adsl-map.ttl#ADSL_TRT01PN>  | "ADSL.TRT01PN"  | xsd:double   |
## | <adsl-map.ttl#ADSL_ITTFL>    | "ADSL.ITTFL"    |              |
## | <adsl-map.ttl#ADSL_SITEGR1>  | "ADSL.SITEGR1"  |              |
## | <adsl-map.ttl#ADSL_SEX>      | "ADSL.SEX"      |              |
## | <adsl-map.ttl#ADSL_ETHNIC>   | "ADSL.ETHNIC"   |              |
## | <adsl-map.ttl#ADSL_COMP8FL>  | "ADSL.COMP8FL"  |              |
## | <adsl-map.ttl#ADSL_RFSTDTC>  | "ADSL.RFSTDTC"  |              |
## | <adsl-map.ttl#ADSL_AGE>      | "ADSL.AGE"      | xsd:double   |
## | <adsl-map.ttl#ADSL_VISNUMEN> | "ADSL.VISNUMEN" | xsd:double   |
## | <adsl-map.ttl#ADSL_EFFFL>    | "ADSL.EFFFL"    |              |
## | <adsl-map.ttl#ADSL_SAFFL>    | "ADSL.SAFFL"    |              |
## | <adsl-map.ttl#ADSL_HEIGHTBL> | "ADSL.HEIGHTBL" | xsd:double   |
## | <adsl-map.ttl#ADSL_MMSETOT>  | "ADSL.MMSETOT"  | xsd:double   |
## | <adsl-map.ttl#ADSL_STUDYID>  | "ADSL.STUDYID"  |              |
## | <adsl-map.ttl#ADSL_RACEN>    | "ADSL.RACEN"    | xsd:double   |
## | <adsl-map.ttl#ADSL_DISONSDT> | "ADSL.DISONSDT" | xsd:double   |
## | <adsl-map.ttl#ADSL_BMIBL>    | "ADSL.BMIBL"    | xsd:double   |
## | <adsl-map.ttl#ADSL_DSRAEFL>  | "ADSL.DSRAEFL"  |              |
## | <adsl-map.ttl#ADSL_AGEU>     | "ADSL.AGEU"     |              |
## | <adsl-map.ttl#ADSL_DURDIS>   | "ADSL.DURDIS"   | xsd:double   |
## | <adsl-map.ttl#ADSL_TRTDUR>   | "ADSL.TRTDUR"   | xsd:double   |
## | <adsl-map.ttl#ADSL_VISIT1DT> | "ADSL.VISIT1DT" | xsd:double   |
## | <adsl-map.ttl#ADSL_SUBJID>   | "ADSL.SUBJID"   |              |
## | <adsl-map.ttl#ADSL_AVGDD>    | "ADSL.AVGDD"    | xsd:double   |
## | <adsl-map.ttl#ADSL_WEIGHTBL> | "ADSL.WEIGHTBL" | xsd:double   |
## | <adsl-map.ttl#ADSL_EDUCLVL>  | "ADSL.EDUCLVL"  | xsd:double   |
## | <adsl-map.ttl#ADSL_SITEID>   | "ADSL.SITEID"   |              |
## | <adsl-map.ttl#ADSL_COMP16FL> | "ADSL.COMP16FL" |              |
## | <adsl-map.ttl#ADSL_RACE>     | "ADSL.RACE"     |              |
## | <adsl-map.ttl#ADSL_TRTEDT>   | "ADSL.TRTEDT"   | xsd:double   |
## | <adsl-map.ttl#ADSL_DISCONFL> | "ADSL.DISCONFL" |              |
```

```
## | <adsl-map.ttl#ADSL_ARM> | "ADSL.ARM" |
## | <adsl-map.ttl#ADSL_DURDSGR1> | "ADSL.DURDSGR1" |
## | <adsl-map.ttl#ADSL_AGEGR1N> | "ADSL.AGEGR1N" | xsd:double
## | <adsl-map.ttl#ADSL_TRT01P> | "ADSL.TRT01P" |
## | <adsl-map.ttl#ADSL_TRT01AN> | "ADSL.TRT01AN" | xsd:double
## | <adsl-map.ttl#ADSL_BMIBLGR1> | "ADSL.BMIBLGR1" |
## -----
```

Then dump the contents of a record in the database.

```
cd ../extdata/sample-rdf
arq --data adsl.ttl --query ADSL-record.rq
```

[illegible]


```
## | <http://www.example.org/datasets/ADSL/01-718-1254> | <http://www.example.org/datasets/vocab/ADSL_I
## | <http://www.example.org/datasets/ADSL/01-718-1254> | <http://www.example.org/datasets/vocab/ADSL_BI
## | <http://www.example.org/datasets/ADSL/01-718-1254> | <http://www.example.org/datasets/vocab/ADSL_RI
## | <http://www.example.org/datasets/ADSL/01-718-1254> | <http://www.w3.org/1999/02/22-rdf-syntax-ns#t
## | <http://www.example.org/datasets/ADSL/01-718-1254> | <http://www.example.org/datasets/vocab/ADSL_S
## | <http://www.example.org/datasets/ADSL/01-718-1254> | <http://www.example.org/datasets/vocab/ADSL_R
## | <http://www.example.org/datasets/ADSL/01-718-1254> | <http://www.example.org/datasets/vocab/ADSL_V
## | <http://www.example.org/datasets/ADSL/01-718-1254> | <http://www.example.org/datasets/vocab/ADSL_A
## | <http://www.example.org/datasets/ADSL/01-718-1254> | <http://www.example.org/datasets/vocab/ADSL_T
## | <http://www.example.org/datasets/ADSL/01-718-1254> | <http://www.example.org/datasets/vocab/ADSL_R
## | <http://www.example.org/datasets/ADSL/01-718-1254> | <http://www.example.org/datasets/vocab/ADSL_A
## | <http://www.example.org/datasets/ADSL/01-718-1254> | <http://www.example.org/datasets/vocab/ADSL_S
## | <http://www.example.org/datasets/ADSL/01-718-1254> | <http://www.example.org/datasets/vocab/ADSL_E
## -----
```

Get underlying data for one cube observation

```
cd ../extdata/sample-rdf
arq --data adsl.ttl --data DC-DEMO-sample.ttl --query DEMOobservations-R-data.rq
```

```
## 12:01:02 WARN riot :: [line: 47, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 48, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 49, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 50, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 51, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 52, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 53, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 66, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 67, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 68, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 69, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 70, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 71, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 72, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 94, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 95, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 96, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 97, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 98, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 99, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 100, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 113, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 114, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 115, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 116, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 117, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 118, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 119, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 132, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 133, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 134, col: 37] Bad IRI: <http://www.example.org/dc/code/
## 12:01:02 WARN riot :: [line: 135, col: 37] Bad IRI: <http://www.example.org/dc/code/
```


[illegible]

186

187

[illegible]

189

[illegible]

191

192

193

194

195

[illegible]

197

198

[illegible]

200


```

## 12:01:02 WARN riot :: [line: 3138, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3139, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3140, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3141, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3142, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3143, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3144, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3170, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3171, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3172, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3173, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3174, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3175, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3176, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3189, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3190, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3191, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3192, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3193, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3194, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3195, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3211, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3212, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3213, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3214, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3215, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3216, col: 37] Bad IRI: <http://www.example.org/dc/cod
## 12:01:02 WARN riot :: [line: 3217, col: 37] Bad IRI: <http://www.example.org/dc/cod
## -----
## | record | p | o |
## =====
## -----

```

How to run this .Rmd file

.. add text ..