PBI

HOW TO WORK WITH PISA IN R

WE WILL SEE

PUBLISHED BY WE WILL SEE

TUFTE-LATEX.GOOGLECODE.COM

First printing, August 2013

Contents

Introduction to PISA

Here there should be an introduction to the PISA data. Who is doing this study and why.

Few words about data structure and availability [1] [2].

[1] The programme for international student assessment (pisa). http://www.oecd.org/pisa/

[2] The organisation for economic co-operation and development (oecd). http://www.oecd.org/

Overview

Key concepts in PISA

Items

Questioneers

Plausible values

BRB replicaes

Where can I find more?

Introduction to R

Very short introduction to R. With references to other materials.

Loading data from Excel or csv files

Info that data is preloaded in PISA packages.
How to head data from Excel
and how to read data from cvs files.

Basic data manipulation

Selecting subset of rows with the [3] function.

[3] `subset()`

Data reshaping
Here there should be an information how to reshape data.
Subselect variables, combine student and school datasets and similar things.

Graphics with ggplot2

How to save figures and other results

Reproducible research

Here the knitr should be introduced.

Where can I find more?

Get your data

In order to work with PISA data in R you need to load the data first. There are at least two way how to do this.

You can download raw data from PISA website [4]. The raw data is available as compressed text files and you can read these files with the `read.fwf()` function. [5]

The second, much easier, approach is to install R package that already consists required data. There are two sets of packages that you may be interested in. Packages with PISA data and packages with supplementary functions that makes it easier to analyse this data set.

[4] The programme for international student assessment (pisa). http://www.oecd.org/pisa/

[5] The `read.fwf()` function is a standard way to read text files in the fixed width format.

R packages with PISA data

Right now there are There are five packages with PISA data. Each package contains data from single PISA study. These packages have following names: `PISA2000lite`, `PISA2003lite`, `PISA2006lite`, `PISA2009lite`, `PISA2012lite`.

Installation of R package requires the download first. Since the datasets are large be prepared to download about 200MB from Internet. But you need to do this only once. per dataset.

In order to install any of these data packages you will need the `devtools` package. In the chapter you will find more details how to install that one.

Suppose that you have the `devtools` package. Than to get data from study PISA 2009 you need to run following commands.

```
library(devtools)
install_github("PISA2009lite", "pbiecek")
```

As a result you shall see an output like that:[6]

[6] Depending on your Internet bandwidth it may take a while.

```
Installing github repo(s) PISA2009lite/master from pbiecek
Downloading PISA2009lite.zip from https://github.com/pbiecek/PISA2009lite/archive/master.zip
Installing package from /var/folders/g3/j8pnss9j3130g4nhj31wxxm0000103/T//RtmptdZ54R/PISA2009lite.zip
Installing PISA2009lite
'/Library/Frameworks/R.framework/Resources/bin/R' --vanilla CMD INSTALL  \
  '/private/var/folders/g3/j8pnss9j3130g4nhj31wxxm0000103/T/RtmptdZ54R/PISA2009lite-master'  \
  --library='/Library/Frameworks/R.framework/Versions/3.0/Resources/library'  \
  --with-keep.source --install-tests

* installing *source* package 'PISA2009lite' ...
** data
*** moving datasets to lazyload DB
** demo
** help
*** installing help indices
** building package indices
** testing if installed package can be loaded
* DONE (PISA2009lite)
```

If there is not `ERROR` in your output it looks like everything went smoothly. The package is installed. In order to work with it you need to load it. Use the `library()` function for that.

```
library(PISA2009lite)
```

You will find five data sets in this package [actually ten, I will explain this later]. These are: data from student questionnaire, school questionnaire, parent questionnaire, cognitive items and scored cognitive items.

```
dim(student2009)
## [1] 515958    437
dim(parent2009)
## [1] 106287     90
dim(school2009)
## [1] 18641   247
dim(item2009)
## [1] 515958    273
```

```
dim(scoredItem2009)
## [1] 515958    227
```

For most of variables in each data set there is a dictionary which decode answers for particular question. Dictionaries for all questions for a given data set are stored as a list of named vectors, these lists are named after corresponding data sets [just add suffix 'dict'].

For example fist six entries in a dictionary for variable CNT in the data set `student2009`.

```
head(student2009dict\$CNT)
##          ALB          ARG          AUS          AUT          AZE
##    "Albania"  "Argentina"  "Australia"    "Austria" "Azerbaijan"
##          BEL
##    "Belgium"
```

Selecting a subset of countries

In some cases you would not work on whole datasets, but only on some subset of countries. You can do this by subseting the dataset. For example, let's take only three countries out of the dataset

```
student2009selected <- subset(student2009, CNT %in% c("ITA", "FRA", "POL"))
dim(student2009selected)
## [1] 40120   437
```

Differences between PISA datasets

There are some differences between different PISA releases.

In PISA 2000 there are three datasets `math2000`, `read2000`, `scie2000` with data from articular area. Different students take different tests, thus these datasets vary in number of rows. All of them contains answers from students questionnaire. In following PISA studies there is a single `student20xx` dataset with outcomes from all areas.

PISAtools: package with supplementary functions

To make it easier to work with PISA data you may use the package `PISAtools`. The installation is similar to the installation of dataset.

```
library(devtools)
install_github("PISAtools", "pbiecek")
library(PISAtools)
```

And the package is ready to use. In next chapters we will show some useful functions that are available there.

Additional datasets

In the `PISAtools` package you will find some additional dataset that might be helpful when working with PISA data. Let's introduce them one by one.

dataset: countryOntology

This ontology was derived from FAO website [7]. It contains information about `211` countries. It may be useful to verify to which group a given country belongs. Let's see columns in this dataset.

[7] Food and agriculture organization of the united nations (fao). `http://www.fao.org/home/en/`

```
head(countryOntology,2)
```

Last column `IS_IN_GROUP` describes to which groups given country belongs. Other columns are just different classifications of particular country.

```
  ISO3 ISO2 UN_CODE UNDP_CODE FAOSTAT_CODE GAUL_CODE FAOTERM_CODE AGROVOC_CODE NAME_EN
1  GRD   GD     308       GRN           86        99        15417         3384 Grenada
2  LBY   LY     434       LIB          124       145        15442         4312   Libya
  LISTNAME_EN
1     Grenada
2       Libya
                                                                          IS_IN_GROUP
1 "FAO_2006,CARICOM_1985,World,CARICOM,CARIFORUM,NFIDC,Americas,Caribbean,FAO,SIDS"
2                       "CEN_SAD,CAEU,World,AMU,northern_Africa,Africa,FAO,COMESA"
```

You can access the information easily with the function `getCountryIdsFromRegion()`. Then you can specify from which region/group of countries you would like to get data. As a result you will get a set of country IDs.

For example, which countries are classified as countries from Western Europe?[8].

[8] These names are case insensitive

```
getCountryIdsFromRegion(region="western_Europe")
## [1] "LUX" "CHE" "DEU" "NLD" "FRA" "MCO" "LIE" "AUT" "BEL"
```

Statistical procedures

Rankings

First we will see how to create ranking for countries. It's easy to do this for all countries, but since there is a lot of them in order to save space we will do this for european countries only.

So first, let's subselect only european countries. The dataset PISAeurope will have the sale variables like student2009 but only rows for students from Europe. [9]

[9] In remaining examples you can replace PISAeurope by student2009 and then you will get ranking for all countries.

```
europe <- getCountryIdsFromRegion(region="Europe")
PISAeurope <- student2009[student2009$CNT %in% europe, ]
```

First let's calculate weighted average performance for all these countries. You can do this with the function `getWeightedAverages()`. You need to specify three arguments. Variables with performance (here Plausible values form Math, Reading and Science), grouping variable (here country) and weights.

As a result you will get data.frame with weighted averages.

```
getWeightedAverages(PISAeurope[,c("PV1MATH", "PV1READ", "PV1SCIE")],
       factor(PISAeurope$CNT), PISAeurope$W_FSTUWT)

     PV1MATH  PV1READ  PV1SCIE
ALB 376.8412 384.8553 390.0552
AUT 495.3798 470.0181 494.3429
BEL 515.6946 506.0709 506.9133
BGR 427.8899 428.7424 439.2365
CHE 535.0264 500.1675 517.0095
CZE 492.5683 478.3270 500.8206
DEU 512.0990 497.2816 520.2056
DNK 503.2260 495.2494 498.9904
```

```
ESP 483.7105 480.9529 488.4244
EST 512.0336 500.3432 527.5742
FIN 540.4180 535.5694 553.6443
FRA 496.7549 495.3661 497.8581
GBR 492.5210 493.9524 513.6889
GRC 465.4492 481.8291 469.3762
HRV 460.6449 475.8268 486.8437
HUN 489.9584 494.2865 502.2476
IRL 487.3271 495.9401 508.2677
ISL 507.3673 500.5733 495.6023
ITA 483.2503 486.3324 489.2809
LIE 533.6883 498.4257 518.5018
LTU 476.4921 467.9608 490.9942
LUX 488.1766 471.1547 483.1795
LVA 481.4847 483.5472 492.8094
MDA 397.3944 388.2060 413.1806
MLT 462.5997 442.1508 461.3761
MNE 402.7139 407.0327 401.5027
NLD 525.8939 508.1992 522.6339
NOR 497.5454 503.0985 499.1366
POL 494.2307 500.1981 507.4326
PRT 487.2701 489.1076 492.8555
ROU 426.4127 424.4139 428.0784
RUS 467.9225 459.4349 478.5926
SRB 442.6190 442.3280 442.8573
SVK 496.7076 477.4750 490.9146
SVN 501.0395 482.7662 511.2549
SWE 493.8699 497.7079 494.8899
```

But initially we were going to derive rankings. It's straight forward. Just use `getRanking()` with same options like for `getWeightedAverages()`.

```
getRanking(PISAeurope[,c("PV1MATH", "PV1READ", "PV1SCIE")],
                  factor(PISAeurope$CNT), PISAeurope$W_FSTUWT)

    PV1MATH PV1READ PV1SCIE
ALB      36      36      36
AUT      14      27      19
BEL       5       3      11
BGR      32      32      32
CHE       2       8       6
```

```
CZE      17      23      13
DEU       6      11       4
DNK       9      14      15
ESP      23      22      25
EST       7       6       2
FIN       1       1       1
FRA      12      13      16
GBR      18      16       7
GRC      28      21      29
HRV      30      25      26
HUN      19      15      12
IRL      21      12       9
ISL       8       5      17
ITA      24      18      24
LIE       3       9       5
LTU      26      28      22
LUX      20      26      27
LVA      25      19      21
MDA      35      35      34
MLT      29      31      30
MNE      34      34      35
NLD       4       2       3
NOR      11       4      14
POL      15       7      10
PRT      22      17      20
ROU      33      33      33
RUS      27      29      28
SRB      31      30      31
SVK      13      24      23
SVN      10      20       8
SWE      16      10      18
```

Note that there is an optional argument `sort`. You can select the column according to which rows should be sorted (by default they are sorted alphabetically).

```
getRanking(PISAeurope[,c("PV1MATH", "PV1READ", "PV1SCIE")],
               factor(PISAeurope$CNT), PISAeurope$W_FSTUWT, sort=1)

    PV1MATH PV1READ PV1SCIE
FIN       1       1       1
```

```
CHE       2       8       6
LIE       3       9       5
NLD       4       2       3
BEL       5       3      11
DEU       6      11       4
EST       7       6       2
ISL       8       5      17
DNK       9      14      15
SVN      10      20       8
NOR      11       4      14
FRA      12      13      16
SVK      13      24      23
AUT      14      27      19
POL      15       7      10
SWE      16      10      18
CZE      17      23      13
GBR      18      16       7
HUN      19      15      12
LUX      20      26      27
IRL      21      12       9
PRT      22      17      20
ESP      23      22      25
ITA      24      18      24
LVA      25      19      21
LTU      26      28      22
RUS      27      29      28
GRC      28      21      29
MLT      29      31      30
HRV      30      25      26
SRB      31      30      31
BGR      32      32      32
ROU      33      33      33
MNE      34      34      35
MDA      35      35      34
ALB      36      36      36
```

Use case - gender differences

Here the distribution of performance between countries and genders can be presented

Different approaches to ranking calculation

- Standard PISA approach via PV
- Percentage of correct answers
- Percentage of correct answers after removing k outliers in items
- Top 90% percentile of the distribution
- Absolute dominance defined as one country over performs other only it also over perform on more than 50% of items

Here there should be an information how to do some simple statistics with the data.

Like weighted regression.

Maybe mixed effect model or generalized mixed effect model.

Data Visualisation

Some examples how to create charts with the use of this data.

Bibliography

Food and agriculture organization of the united nations (fao). http://www.fao.org/home/en/.

The organisation for economic co-operation and development (oecd). http://www.oecd.org/.

The programme for international student assessment (pisa). http://www.oecd.org/pisa/.