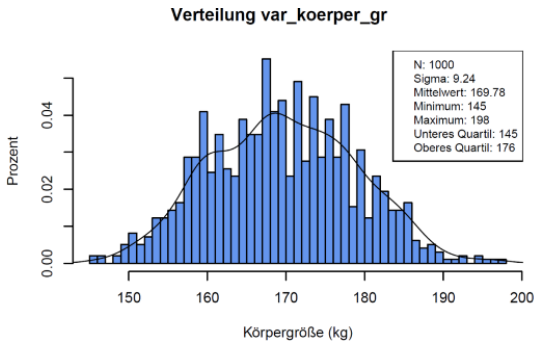
Eine grundlegende Prüfung der Datenqualität von epidemiologischen Forschungsdaten mit MOQA

Letzte Änderung am 2017/06/20

Hintergrund

Jedes epidemiologische Forschungsprojekt, das Datenerhebungen durchführt, steht vor der Herausforderung, die Qualität der Daten kontinuierlich zu prüfen.

Um grundlegende Verfahren zur Plausibilitätsprüfung von Daten, ohne Kenntnis von Einheiten, Wertebereichen und Codierungen der entsprechenden Variablen, anwenden zu können, liegt der Fokus der bereitgestellten Skript-Bibliothek auf der allgemeingültigen Generierung von Reports. Auf diese Weise können Aussagen zur Verteilung von Häufigkeiten, zur Vollzähligkeit und zur Vollständigkeit der Daten getroffen werden.



Warum gerade R?

Die Open Source Statistiklösung R kann kostenfrei genutzt werden und ist bedingt durch die große Nutzer-Community sehr ausführlich dokumentiert. Der geringe Einarbeitungsaufwand in RStudio und das unkomplizierte Integrieren neuer Bibliotheken führen schnell zu den gewünschten Erfolgen. Zahlreiche Publikationen zum Thema werden zudem in einem fachspezifischen Journal („the R Journal") veröffentlicht.

Welche Alternativen gibt es?

Es gibt natürlich alternative Statistiklösungen, die ebenfalls etabliert und gut dokumentiert sind, wie SAS, STATA und SPSS. Diese werden ergänzt um interaktive und intuitiv nutzbare Lösungen, wie QLIK-Sense oder JMP (Statistical Discovery with SAS).

Vor allem letztere machen ungemein Spaß in der Anwendung, verursachen aber mitunter hohe Kosten in Anschaffung und/oder Betrieb. Für Kohortenstudien und Register mit nur geringen finanziellen Mitteln, sind diese kostenpflichtigen Statistiklösungen typischerweise nur bedingt von Interesse.

Ein allgemeiner Lösungsansatz

Ziel der Skript-Bibliothek ist es, die Qualität der Daten möglichst generisch je Studienvariable mit R zu visualisieren. Dies geschieht vorwiegend über:

- die Häufigkeitsanalyse gültiger Werte und Missings,
- die Verteilung der Daten, sowie
- die Unterscheidung in kategoriale und metrische Daten.

Dies erlaubt allgemeine Berichte zu generieren und entsprechende Aussagen abzuleiten. Für konkretere Aussagen sind Kenntnisse über Metadaten (z.B. Variablenbeschreibung, Einheit) und Codierungen (z.B. gültige Antworten, Missings) der Variablen erforderlich.

Kategoriale Daten

Kategoriale Daten entstehen beispielsweise bei Auswahlfragen (z.B. für Antwortoptionen: ja, nein, vielleicht, weiß nicht). Die Darstellung der Häufigkeitsverteilung in grafischer und tabellarischer Form erlaubt favorisierte Antworten und die Anteile von Missings schnell zu identifizieren (vgl. Beispiel-Report1).

Die Skript-Bibliothek ermöglicht die **Angabe von Code-Listen** für kategoriale Daten. Wird eine Code-Liste angegeben, werden die entsprechenden Graphen und Tabellen individuell beschriftet. Wird keine Code-Liste angegeben, wird basierend auf dem konfigurierten **Schwellwert zur Erkennung von Missings** (standardmäßig werden Werte > 99000 als Missing betrachtet), in gültige Werte und Missings unterschieden.

Metrische Daten

Metrische Daten werden beispielsweise bei Messungen (Körpergröße, Gewicht, etc.) erzeugt. Diese Daten werden mithilfe der Skript-Bibliothek tabellarisch und in Form von Verteilungs- bzw. Wahrscheinlichkeitsplots dargestellt (vgl. Beispiel-Report2).

- Die **Histogramm-Darstellung** zeigt die Verteilung der Werte und gibt Mittelwerte, Standardabweichung, etc. an.
- Die **tabellarische Übersicht** gibt Aufschluss über das Verhältnis gültiger Werte und Missings.
- Der **Box-Whisker Plot** zeigt Quantile und hilft Ausreißer im Datenbestand zu identifizieren.
- Der **Quantil-Quantil-Plot** (auch QQ-Plot) zeigt die Abweichung der Daten zur Normalenverteilung. Die Daten sind zur Beantwortung der wiss. Fragestellung geeignet, wenn sich beide Linien möglichst annähern (d.h. der Test ist aus epidemiologischer Sicht signifikant).

Lernen am Beispiel

Der Beispielreport für metrische Daten konnte unter Verwendung der Skript-Bibliothek, wie im nachfolgenden R-Schnipsel gezeigt, mit nur wenigen Aufrufen realisiert werden.

```
# specify the csv import file with metric data, use one column per variable, first row should contain variable name, following rows should contain data
metric_datafile='c:/mosaic/sample_data/metric_single_var.csv'

#specify output folder
outputFolder="c:/mosaic/output/"

# load MOQA Functions from library
library(MOQA)

#set missings threshold, optional, default is 99900
mosaic.setGlobalMissingTreshold(99900)

#set variable unit, optional
mosaic.setGlobalUnit("(kg)")

#set variable description, optional
mosaic.setGlobalDescription("Körpergröße")

#create PDF-report
mosaic.createSimplePdfmetric(metric_datafile, outputFolder)
```

Nutzen Sie eines der mitgelieferten Beispielskripte zur Graphengenierung für metrische und kategoriale Daten bzw. einzelne oder mehrere Variablen als Arbeitsgrundlage und passen Sie diese nach

Belieben an.

Funktionsübersicht

Die nachfolgende Tabelle (englisch) listet in der Skript-Bibliothek enthaltene Funktionen (Version 2.0).

Function	Description
mosaic.info()	about this library
mosaic.setGlobalMissingTreshold(threshold)	Set Global Threshold for Missings, e.g. 99000
mosaic.setGlobalUnit(unit)	Set Global Unit Label to be used in graphs, e.g. "(cm)"
mosaic.setGlobalDescription(description)	Set Global Description for variable data, e.g. „waist circumference“
mosaic.loadCsvData(filename)	Load CSV Data from file, e.g. „c:\data.csv“
mosaic.importToolboxSpssDataFile(filename)	Import data from >Toolbox for Research< SPSS-Export Dat-File to dataframe
mosaic.countValue(searchvalue,datacolumn)	Count occurrence of search value in data column
mosaic.preProcessMetricData(data)	Pre-process metric data to allow missing-ratio table
mosaic.preProcessCategoricalData(data)	Identify unique values in data column, get abs, perc and cumulative stats
mosaic.generateMetricTablePlot(data, num of columns, column index, varname)	Generate missing-ratio table for metric data
mosaic.generateMetricPlots(data snippet, varname)	Generate graphs for metric data
mosaic.beginPlot(varname)	Begin plotting, generate PDF-File with given variable name
mosaic.addFootnote()	Add a Footnote with timestamp and MOSAIC text
mosaic.finishPlot()	Finish plotting, close PDF File
mosaic.setGlobalCodelist(codelist)	Set and parse a global code list for categorical data, e.g. c("1=yes","2=no","99996=no information")
mosaic.generateCategoricalPlot(dataframe, varname)	Create plots for categorical data
mosaic.createSimplePdfCategorical(inputfile, outputfolder)	Create simple pdf file for categorical data using the functions listed above
mosaic.createSimplePdfCategoricalDataframe(dataframe, outputfolder)	Create simple pdf file for categorial dataframe
mosaic.createSimplePdfMetric(inputfile, outputfolder)	Create simple pdf file for metric data using the functions listed above
mosaic.createSimplePdfMetricDataframe(dataframe, outputfolder)	Create simple pdf file for metric data frame
mosaic.getTimestamp()	Get formatted timestamp, e.g. 2015_09_16_235811

Installation der Skript-Bibliothek

Technischer Rahmen

Die Skript-Bibliothek wurde unter Verwendung von **R in der Version 3.4.0** realisiert. Diese können Sie hier herunterladen.

RStudio (Desktop Version 0.99.484) ist für die Nutzung der Skript-Bibliothek nicht erforderlich, vereinfacht aber die Anwendung von R. Es wird kostenfrei unter folgendem Link bereitgestellt: https://www.rstudio.com/products/rstudio/download/

Bei Aufruf der Skript-Bibliothek werden 4 benötigte R-Pakete, falls nicht vorhanden, automatisch installiert. Dieser Schritt erfordert Internetzugriff.

Einrichtung des R-Packages 'MOQA'

Die Bereitstellung der R-Bibliothek erfolgt **kostenfrei unter AGPLv3-Lizenz**. Dies reduziert den Einrichtungsaufwand auf ein Minimum und die umfangreiche Begleitdokumentation erlaubt einen einfachen Einstieg. Beispielskripte und Datenbeispiele werden natürlich weiterhin über die MOSAIC-Projekt-Homepage bereitgestellt.

Mehr Informationen zum MOQA-Package direkt im CRAN-Repository und in der entsprechenden Publikation bei METHODS OF INFORMATION IN MEDICINE (hier noch unter dem alten Namen mosaicQA)

Folgender Aufruf installiert das MOQA-Package und die erforderlichen Abhängigkeiten:

```
#installation
install.packages("MOQA")

#load library
library(MOQA)
```

FAQ

Ist Excel ein geeignetes Werkzeug für die Verwaltung von Daten in einer multizentrischen Studie?

Kontakt

Die Bibliothek und hier zur Verfügung gestellte Skripte sind kommentiert. Sollten dennoch Fragen auftreten oder sollten Sie Anmerkungen haben, zögern Sie bitte nicht uns zu kontaktieren. Wir freuen uns über Ihr Feedback. Nutzen Sie unser **Kontaktformular** oder wenden Sie sich an einen der nachfolgenden **Ansprechpartner**:

Martin Bialke martin.bialke@uni-greifswald.de, Tel.: +49 3834 86 7580