Reproducible codes

# Informationen

## Was Sind „Reproducible codes“?

Reproducible Codes meint die Dokumentation und öffentliche Zurverfügungstellung der Codes, die aus den Daten eines Projekts (optimalerweise den Rohdaten) die in einem Paper berichteten Ergebnisse produzieren. Reproducible Codes ermöglichen damit eine genaue und umfassende Kommunikation der eigenen Analysen, ein exaktes Nachvollziehen und eine angemessene Interpretation dieser Analysen und der darauf aufbauenden Ergebnisse und eine direkte Replikation des analytischen Vorgehens.

Konkret beinhaltet „Reproducible Codes“:

1. die Dokumentation aller Datenaufbereitungen und statistischen Analysen in einer oder mehreren verständlich kommentierten Syntax-Datei(en)
2. die Zurverfügungstellung dieser Dateien auf einem öffentlich zugänglichen Online-Repositorium
3. Die Verlinkung zu diesen Dateien im Paper

## Links zu weiterführenden Informationen

* <https://www.rstudio.com/wp-content/uploads/2015/06/rmarkdown-german.pdf>
* <https://google.github.io/styleguide/Rguide.xml>
* <https://github.com/rubenarslan/repro_web_stack>
* <http://robinlovelace.net/r/2014/07/15/naming-conventions-r.html>

# Lösungen

## Reproducible codes Kurz und knapp

In einigen Fällen werden die Codes der statistischen Analysen evtl. nicht optimal aufbereitet und kommentiert sein und evtl. haben Sie nicht mehr ausreichend Zeit dies nachzuholen. Um in diesem Fall trotzdem zeitnah Reproducible Codes zur Verfügung zu stellen, sollte zumindest geprüft werden, ob die Anwendung der Codes auf die Daten tatsächlich die im Paper berichteten Zahlen erzeugt. Darüber hinaus kann knapp geprüft werden, wo einfache Kommentierungen ergänzt werden können, um die Verständlichkeit der Codes zu erhöhen. Um auch in kurzer Zeit hilfreiche Codes zur Verfügung zu stellen, kann bspw. geprüft werden, ob folgende Kriterien zutreffen:

Die Codes sollten in einem öffentlichen und frei zugänglichen online-Repositorium (bspw. [www.osf.io](http://www.osf.io)) gespeichert werden. Im Paper (bspw. zu Beginn des Statistical Analyses oder des Results-Teils) sollte der Link zu den Codes gegeben werden. Optimalerweise stehen dort auch die Daten als Open Data zur Verfügung, so dass die Ergebnisse eines Papers tatsächlich direkt nachvollzogen werden können und ggf. alternative Analysen durchgeführt werden können. Aber selbst wenn die Daten (bspw. aus rechtlichen Gründen oder wegen der fehlenden Möglichkeit der Anonymisierung) nicht frei zugänglich gemacht werden können, helfen Reproducible Codes auch alleine, um die statistischen Analysen (bspw. die exakte Spezifikation komplexerer Modelle) exakt nachvollziehen zu können.

## REPRODUCIBLE CODES Detailliert

Optimalerweise wird bereits vor der Datenerhebung geplant ein Codebuch zu führen, die Variablen bei/nach der Datenerhebung entsprechend zu benennen und während der Datenanalyse ein hierzu passendes reproduzierbares und verständliches Skript zu führen.

Die folgenden Tipps und Richtlinien zur Erstellung eines übersichtlichen und nachvollziehbaren Codes haben beispielhaften Charakter und sind für die Statistikumgebung R formuliert.

**Gliederung des Skripts**

Empfohlen wird, eine einfache Gliederung im Skript einzuhalten. Diese kann im Einzelfall auch von empfohlenen Standards abweichen, sollte jedoch eine nachvollziehbare Struktur enthalten. Hier ein Beispiel:

**Variablen und Funktionen benennen und festhalten**

Die Benennung von Variablen und Funktionen könnte nach diesem einfachen Schema erfolgen:

**Richtlinien zum Kommentieren**

* Kommentieren Sie Ihren Code.
* Ganze Kommentarzeilen, in denen kein Befehl enthalten ist beginnen mit # und anschließendem Leerzeichen.
* Kurze Kommentare, die Teile eines Befehls oder einer Funktion beschreiben, beginnen in der gleichen Zeile wie der Befehlsteil – nach zwei Leerzeichen und # .

# Create histogram of frequency of campaigns by pct budget spent.

hist(df$pct.spent,

breaks = "scott", # method for choosing number of buckets

main = "Histogram: fraction budget spent by campaignid",

xlab = "Fraction of budget spent",

ylab = "Frequency (count of campaignids)")

**Richtlinien innerhalb der Syntax**

Auch innerhalb der Syntax sollte einigermaßen Ordnung herrschen. Diese Punkte haben sich etabliert und werden von vielen Anwendern beachtet.

* Nicht mehr als 80 Zeichen pro Zeile
* Vor und nach Sonderzeichen in Befehlen (+, -, <, <-, usw.) je ein Leerzeichen
* Kein Leerzeichen vor Kommas aber danach

tab.prior <- table(df[df$days.from.opt < 0, "campaign.id"])

total <- sum(x[, 1])

total <- sum(x[1, ])

* Doppeltes Leerzeichen nach einem Befehl
* { bleibt in der gleichen Zeile wie der Befehl, } endet in eigener Zeile
* Für Zuweisungen wird <- benutzt, nicht =

if (is.null(ylim)) {

ylim <- c(0, 0.06)

}

* Komplexe Befehle sollen auf mehrere Zeilen verteilt werden.
* else-Argumente befinden sich in der gleichen Zeile wie } und { .
* Kein Semikolon verwenden.

if (condition) {

one or more lines

} else {

one or more lines

}

# Ein EINFACHES BEISPIEL

Das folgende Beispiel ist bewusst einfach gehalten und soll verdeutlichen, dass es v.a. darauf ankommt, jeden Schritt den man in R oder einem anderen Programm tätigt, zu kommentieren um ein nachvollziehbares Skript zu erstellen.

