Einführung in die statistische Datenanalyse mit R

Aufgabenblatt – Musterlösung

David Benček

Wintersemester 2015/16

Aufgabe 2

Verschaffen Sie sich einen Überblick zu den im Aufsatz analysierten Daten **mithilfe von R**.

```
buhaug <- read.csv("./data/aufgabenblatt/buhaug_gates.csv", stringsAsFacto
```

2a

Wieviele Beobachtungen umfasst der Datensatz?

2a

Wieviele Beobachtungen umfasst der Datensatz?

```
dim(buhaug)
```

```
## [1] 265  65
```

2b

Welcher Zeitraum wird von den Daten abgedeckt?

2b

Welcher Zeitraum wird von den Daten abgedeckt?

```
#summary(buhaug$begin)
#min(buhaug$begin)
#max(buhaug$begin)
range(buhaug$begin)
```

```
## [1] 1946 2000
```

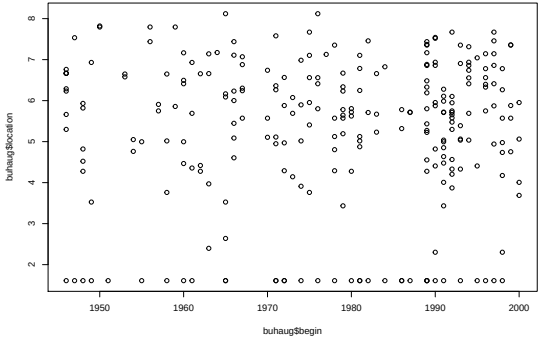
2c

Erstellen Sie einen Scatterplot für jede der abhängigen Variablen und zeigen Sie damit ihre Entwicklung in Abhängigkeit vom Konfliktbeginn. Können Sie einen Trend erkennen?

2c

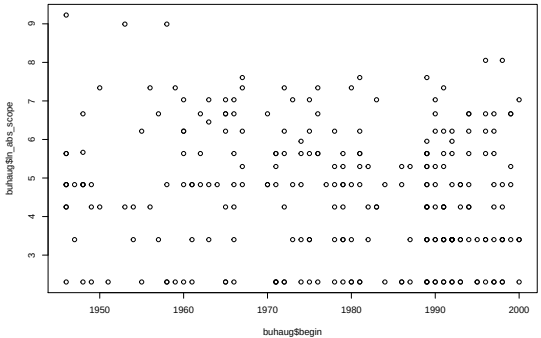
Erstellen Sie einen Scatterplot für jede der abhängigen Variablen und zeigen Sie damit ihre Entwicklung in Abhängigkeit vom Konfliktbeginn. Können Sie einen Trend erkennen?

```
plot(buhaug$begin, buhaug$location)
```



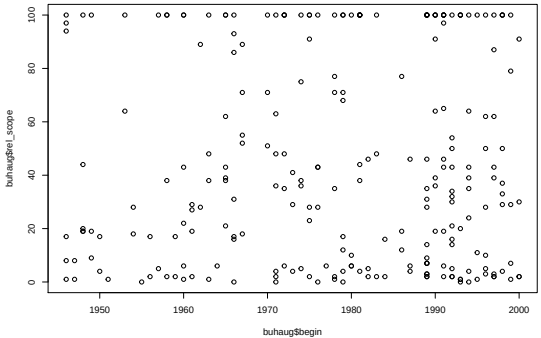
2c

```
plot(buhaug$begin, buhaug$ln_abs_scope)
```



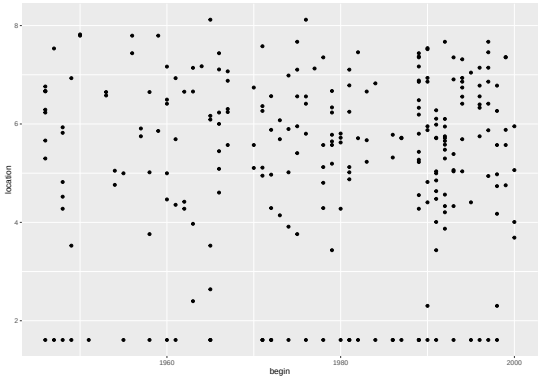
2c

```
plot(buhaug$begin, buhaug$rel_scope)
```



2c Alternative I

```
library(ggplot2)
ggplot(buhaug, aes(begin, location)) +
  geom_point()
```



2c Alternative II

```
library(dplyr); library(tidyr)
dep_variables <- buhaug %>%
  select(begin, location, ln_abs_scope, rel_scope) %>%
  gather(variable, value, -begin)
ggplot(dep_variables, aes(begin, value, colour = variable)) +
  geom_point(alpha=0.6, position = "jitter")
```

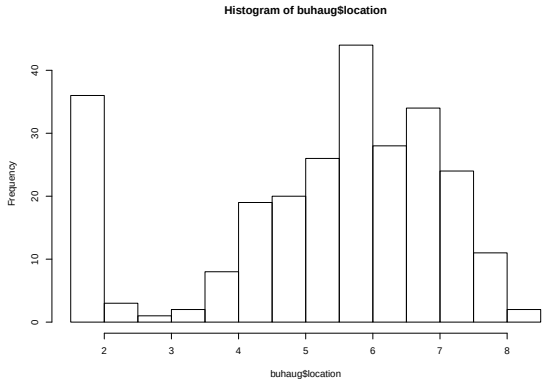
2d

Plotten Sie Histogramme der abhängigen Variablen.

2d

Plotten Sie Histogramme der abhängigen Variablen.

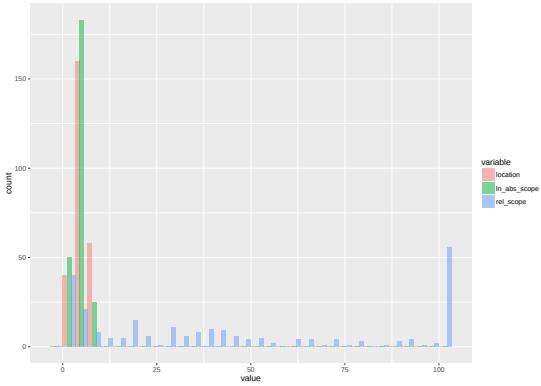
```
hist(buhaug$location)
```




```
#hist(buhaug$ln_abs_scope)
#hist(buhaug$rel_scope)
```

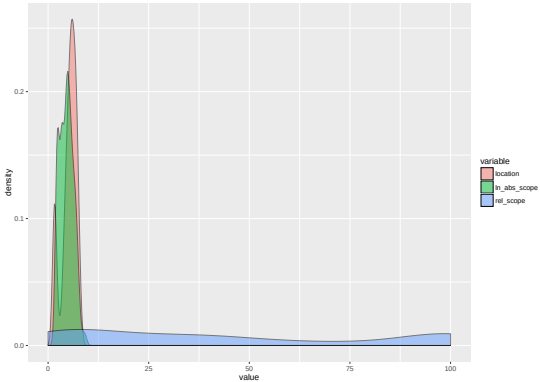
2d Alternative

```
ggplot(dep_variables, aes(value, fill = variable)) +
  geom_histogram(alpha=0.5, position = "dodge")
```



2d stetige Variante

```
ggplot(dep_variables, aes(value, fill = variable)) +
  geom_density(alpha=0.5, position = "dodge")
```



3a

Vollziehen Sie die deskriptiven Statistiken der Tabelle I im Aufsatz von Buhaug/Gates nach und berechnen Sie alle angegebenen Werte.

3a

Vollziehen Sie die deskriptiven Statistiken der Tabelle I im Aufsatz von Buhaug/Gates nach und berechnen Sie alle angegebenen Werte.

```
descriptive_stats_df <- buhaug[, c("location", "ln_abs_scope", "rel_scope"
                                   "ln_land_area", "identity",
                                   "incompatibility","duration",
                                   "border", "resource",
                                   "mountain", "forest")]

# mit dplyr:
descriptive_stats_df <- select(buhaug, location, ln_abs_scope, rel_scope,
                               ln_land_area, identity, incompatibility,
                               duration,  border, resource, mountain,
                               forest)
```

3a

```r
# einzeln:
sum(!is.na(descriptive_stats_df$location))
```

```
## [1] 258
```

```r
mean(descriptive_stats_df$location, na.rm = TRUE)
```

```
## [1] 5.233605
```

```r
sd(descriptive_stats_df$location, na.rm = TRUE)
```

```
## [1] 1.823652
```

```r
min(descriptive_stats_df$location, na.rm = TRUE)
```

```
## [1] 1.609438
```

3a Alternative

```r
# spaltenweise:
Observations <- sapply(descriptive_stats_df, function(x){
  sum(!is.na(x))
})
Mean <- sapply(descriptive_stats_df, function(x){
  mean(x, na.rm = TRUE)
})
SD <- sapply(descriptive_stats_df, function(x){
  sd(x, na.rm = TRUE)
})
Min <- sapply(descriptive_stats_df, function(x){
  min(x, na.rm = TRUE)
})
Max <- sapply(descriptive_stats_df, function(x){
  max(x, na.rm = TRUE)
})
```

3b

Erstellen Sie einen *data frame*, der exakt Tabelle I entspricht.

3b

Erstellen Sie einen *data frame*, der exakt Tabelle I entspricht.

```
table_1 <- data.frame(Observations, Mean, SD, Min, Max)
table_1 <- round(table_1, 2)
```

Aufgabe 4

Tabelle II zeigt eine Matrix aller paarweisen Korrelationskoeffizienten der betrachteten Variablen.

- Was sagt Ihnen ein einzelner Wert in Tabelle II?
- Was sagt der Korrelationskoeffizient von “Identity” und “Incompatibility” aus? Interpretieren Sie kurz.

4c

Der Befehl `cor()` berechnet die Korrelation zweier Variablen. Wenn Sie ihn auf einen *data frame* anwenden, erhalten Sie alle paarweisen Korrelationskoeffizienten. Rechnen Sie damit Tabelle II nach. (Tips: (i) Erstellen Sie zunächst einen *data frame*, der nur die benötigten Variablen enthält. (ii) Das Argument `use` ist hilfreich, lesen Sie mehr darüber auf der Hilfeseite von `cor()`)

4c

Der Befehl `cor()` berechnet die Korrelation zweier Variablen. Wenn Sie ihn auf einen *data frame* anwenden, erhalten Sie alle paarweisen Korrelationskoeffizienten. Rechnen Sie damit Tabelle II nach. (Tips: (i) Erstellen Sie zunächst einen *data frame*, der nur die benötigten Variablen enthält. (ii) Das Argument `use` ist hilfreich, lesen Sie mehr darüber auf der Hilfeseite von `cor()`)

```
cor_matrix_df <- buhaug[, c("location", "ln_abs_scope", "rel_scope",
                            "ln_land_area", "identity", "incompatibility",
                            "duration", "border", "resource",
                            "mountain", "forest")]
# mit dplyr
cor_matrix_df <- select(buhaug, location, ln_abs_scope, rel_scope,
                        ln_land_area, identity, incompatibility,
                        duration,  border, resource, mountain,
                        forest)
```

4c

```
table_2 <- as.data.frame(cor(cor_matrix_df, use="complete.obs"))
table_2 <- round(table_2, 3)
table_2[upper.tri(table_2, diag = TRUE)] <- NA

table_2
```

```
##                location ln_abs_scope rel_scope ln_land_area identity
## location             NA           NA        NA           NA       NA
## ln_abs_scope      0.600           NA        NA           NA       NA
## rel_scope         0.008        0.485        NA           NA       NA
## ln_land_area      0.536        0.438    -0.457           NA       NA
## identity          0.543        0.268    -0.058        0.275       NA
## incompatibility  -0.548       -0.094     0.305       -0.314   -0.596
## duration          0.153        0.247     0.199        0.037    0.191
## border            0.349        0.328     0.174        0.116    0.315
## resource          0.109        0.269     0.102        0.140    0.053
## mountain         -0.006       -0.010     0.024       -0.037    0.077
```

Aufgabe 5

Replizieren Sie die Regressionsmodelle 1 bis 6. Betrachten Sie jeweils die Modellzusammenfassungen und geschätzten Koeffizienten. Beschreiben Sie kurz, was Sie über den Einfluss der einzelnen unabhängigen Variablen erfahren.

Aufgabe 5

Replizieren Sie die Regressionsmodelle 1 bis 6. Betrachten Sie jeweils die Modellzusammenfassungen und geschätzten Koeffizienten. Beschreiben Sie kurz, was Sie über den Einfluss der einzelnen unabhängigen Variablen erfahren.

```
# location
model_1 <- lm(location ~ ln_abs_scope + ln_land_area + identity +
                incompatibility,
              data = buhaug)
#summary(model_1)

model_2 <- lm(location ~ rel_scope + ln_land_area + identity +
                incompatibility,
              data = buhaug)
#summary(model_2)
```

Aufgabe 5

```
# absolute scope
model_3 <- lm(ln_abs_scope ~ location + ln_land_area + duration +
                border + resource,
              data = buhaug)
#summary(model_3)

model_4 <- lm(ln_abs_scope ~ location + ln_land_area + duration +
                border + resource +
                mountain + forest,
              data = buhaug)
#summary(model_4)
```

Aufgabe 5

```
# relative scope
model_5 <- lm(rel_scope ~ location + ln_land_area + duration +
                border + resource,
              data = buhaug)
#summary(model_5)

model_6 <- lm(rel_scope ~ location + ln_land_area + duration +
                border + resource +
                mountain + forest,
              data = buhaug)
#summary(model_6)
```

Aufgabe 5 (Zusatz)

```
# nice output
library(stargazer)

stargazer(list(model_1, model_2), type="html", out = "./table_3.html")

stargazer(list(model_3, model_4, model_5, model_6),
          type="html", out = "./table_4.html")
```

Aufgabe 6

Hypothese: *Eine bessere Infrastruktur begünstigt die Ausweitung von Konflikten innerhalb eines Landes.*

Verwenden Sie als Approximation der Infrastrukturqualität das BIP pro Kopf zu Beginn jedes Konflikts im Datensatz. Finden Sie die notwendigen Daten in einer Datenbank Ihrer Wahl (mögliche Quellen sind: Weltbank, IWF, OECD), verknüpfen Sie die Informationen zum BIP pro Kopf mit dem Datensatz und schätzen Sie die Modelle aus Aufgabe 5 nochmals mit der neuen Variable. Beschreiben Sie kurz ihre Erkenntnis zum Einfluss vom BIP pro Kopf und treffen Sie eine Aussage zur obenstehenden Hypothese.

Hinweise: (i) Das im Seminar gezeigte Paket `countrycode` ist hilfreich, um die einzelnen Länder zu identifizieren und aus unterschiedlichen Quellen zusammenzuführen. (ii) Reichen Sie bei Abgabe Ihrer Lösung auch Ihre Rohdaten aus der Datenbank ein.

Aufgabe 6

- Datenquelle: `https://www.clio-infra.eu/datasets/search`
- gefunden auf: `ourworldindata.org/data/growth-and-distribution-of-prosperity/gdp-data/`

```
library(countrycode)
gdp <- read.csv("./data/aufgabenblatt/clioinfra.csv",
                stringsAsFactors = FALSE)
buhaug <- gdp %>%
  filter(!is.na(ccode)) %>%
  select(ccode, country.name, starts_with("X")) %>%
  gather(year, gdp, -c(ccode, country.name)) %>%
  mutate(year = as.integer(gsub("X", "", year)),
         cow = countrycode(ccode, "iso3n", "cown")) %>%
  select(year, cow, gdp) %>%
  right_join(buhaug, c("cow" = "cow", "year" = "begin"))
```

Aufgabe 6

```
# location
model_1b <- lm(location ~ ln_abs_scope + ln_land_area + identity +
                 incompatibility + log(gdp),
               data = buhaug)
#summary(model_1b)

model_2b <- lm(location ~ rel_scope + ln_land_area + identity +
                 incompatibility + log(gdp),
               data = buhaug)
#summary(model_2b)
```

Aufgabe 6

```
# absolute scope
model_3b <- lm(ln_abs_scope ~ location + ln_land_area + duration +
                 border + resource + log(gdp),
               data = buhaug)
#summary(model_3b)

model_4b <- lm(ln_abs_scope ~ location + ln_land_area + duration +
                 border + resource +
                 mountain + forest + log(gdp),
               data = buhaug)
#summary(model_4b)
```

Aufgabe 6

```
# relative scope
model_5b <- lm(rel_scope ~ location + ln_land_area + duration +
                 border + resource + log(gdp),
               data = buhaug)
#summary(model_5b)

model_6b <- lm(rel_scope ~ location + ln_land_area + duration +
                 border + resource +
                 mountain + forest + log(gdp),
               data = buhaug)
#summary(model_6b)
```