Problem Set 1 - Antworten

Christoph Nguyen

5/11/2020

# Setup

## Setup Packages

Wir arbeiten in diesem Problem Set mit den Daten aus “Ehlert, Niek, Annika Hennl, and André Kaiser. 2007. “Föderalismus, Dezentralisierung Und Performanz. Eine Makroquantitative Analyse Der Leistungsfähigkeit Territorialer Politikorganisation in Entwickelten Demokratien.” Politische Vierteljahresschrift 48(2): 243–68."

Bevor wir mit den inhaltlichen Fragen beginnen, müssen wir aber zuerst die notwendigen Packete installieren und aktivieren.

Falls ihr die Packete/libraries noch nicht installiert habt, müssen dieser erst installiert werden. Das müsst ihr nur einmal machen.

install.packages("tidyverse", repos = "http://cran.us.r-project.org")

## Installing package into 'C:/Users/cgngu/Documents/R/win-library/4.0'  
## (as 'lib' is unspecified)

## package 'tidyverse' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\cgngu\AppData\Local\Temp\RtmpiyHWC6\downloaded\_packages

Wenn das Paket installiert ist, müsst ihr es auch noch aktivieren. Das müsst ihr jedesmal machen.

library(tidyverse)

## -- Attaching packages ---------------------------------------------------------------------------------------------------------------------------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.0 v purrr 0.3.4  
## v tibble 3.0.1 v dplyr 0.8.5  
## v tidyr 1.0.2 v stringr 1.4.0  
## v readr 1.3.1 v forcats 0.5.0

## -- Conflicts ------------------------------------------------------------------------------------------------------------------------------------------------------------- tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

## Daten Laden

Nun müssen wir die Daten einladen. Wir beziehen die Daten hier direkt aus dem Internet. Allerdings könnt ihr auch die Daten von eurer Festplatte einlesen.

data\_fed<-read\_csv("https://raw.githubusercontent.com/cgnguyen/parlament\_in\_r/master/fd-cross.csv")

## Parsed with column specification:  
## cols(  
## .default = col\_double(),  
## country = col\_character(),  
## label = col\_character()  
## )

## See spec(...) for full column specifications.

# Frage 1

Berechne für die Sozialausgaben (socexp10) und die Inflationsrate (infl10) die folgenden deskriptiven Werte.

1. Durschnitt
2. Mittelwert
3. Minimalwert
4. Maximalwert

Mit base R

summary(data\_fed[,c("socexp10","infl10")])

## socexp10 infl10   
## Min. :0.06566 Min. :1.000   
## 1st Qu.:0.08909 1st Qu.:1.018   
## Median :0.12991 Median :1.025   
## Mean :0.13280 Mean :1.058   
## 3rd Qu.:0.15780 3rd Qu.:1.037   
## Max. :0.28191 Max. :1.689   
## NA's :8

Im Tidyverse

data\_fed %>%  
 select(socexp10,infl10)%>%  
 summary(.)

## socexp10 infl10   
## Min. :0.06566 Min. :1.000   
## 1st Qu.:0.08909 1st Qu.:1.018   
## Median :0.12991 Median :1.025   
## Mean :0.13280 Mean :1.058   
## 3rd Qu.:0.15780 3rd Qu.:1.037   
## Max. :0.28191 Max. :1.689   
## NA's :8

Ihr könnt auch spezifische Werte ansteuern, in dem ihr die Befehle “mean”, “median” etc. nutzt.

# Frage 2

Berechne Mittelwert und Durchschnitt für föderale und zentralisierte Staaten (feddummy).

Hier nutzen wir die *group\_by* Funktion aus dem tidyverse. Wir solten hier auch feddummy in eine wirkliche Dummyvariable umwandeln.

data\_fed<-  
 data\_fed %>%   
 mutate(feddummy = factor(recode(feddummy,  
 `0`="Nein",  
 `1`="Ja")))  
  
 table\_summary\_2<-  
 data\_fed %>%  
 group\_by(feddummy)%>%  
 summarize(mean\_socexp10= mean(socexp10,na.rm=T),  
 median\_socexp10= median(socexp10,na.rm=T),  
 mean\_infl10= mean(infl10,na.rm=T),  
 median\_infl10= median(infl10,na.rm=T))  
   
 table\_summary\_2

## # A tibble: 2 x 5  
## feddummy mean\_socexp10 median\_socexp10 mean\_infl10 median\_infl10  
## <fct> <dbl> <dbl> <dbl> <dbl>  
## 1 Ja 0.0957 0.0774 1.04 1.02  
## 2 Nein 0.150 0.147 1.07 1.03

Das sieht aber ziemlich hässlich aus. Um die Darstellung von Ergebnissen in Rmarkdown etwas schöner zu gestalten, nutze ich hier Funktionen “kable” aus dem Packet *knitr::kable()* .Schaut mal in die orginalle Rmarkdown Datei, um zu sehen, wie das in der Praxis funktioniert.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| feddummy | mean\_socexp10 | median\_socexp10 | mean\_infl10 | median\_infl10 |
| Ja | 0.0956749 | 0.0774453 | 1.035492 | 1.018271 |
| Nein | 0.1501189 | 0.1467336 | 1.065832 | 1.030074 |

# Frage 3

Berechne Mittelwert und Durchschnitt für Staaten mit und ohne englischer Rechtskultur

Die Logik ist hier genau die gleich. Wir können in das Papier oder die Variablenliste schauen, um den Namen der Variable zu identifizieren. Der Name der Variable ist “legeng”.

data\_fed<-  
 data\_fed %>%   
 mutate(legeng = factor(recode(legeng,  
 `0`="Nein",  
 `1`="Ja")))  
  
 table\_summary\_3<-  
 data\_fed %>%  
 group\_by(legeng)%>%  
 summarize(mean\_socexp10= mean(socexp10,na.rm=T),  
 median\_socexp10= median(socexp10,na.rm=T),  
 mean\_infl10= mean(infl10,na.rm=T),  
 median\_infl10= median(infl10,na.rm=T))

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| legeng | mean\_socexp10 | median\_socexp10 | mean\_infl10 | median\_infl10 |
| Ja | 0.1031562 | 0.0880133 | 1.022739 | 1.022131 |
| Nein | 0.1415133 | 0.1380478 | 1.066492 | 1.027222 |

# Frage 4

Berechne Mittelwert und Durchschnitt für föderale und zentralisierter Staaten mit und ohne englischer Rechtskultur

Der Befehl *dplyr::group\_by*kann auch auf mehrere Variablen angewandt werden.

table\_summary\_4<-  
 data\_fed %>%  
 group\_by(feddummy,legeng)%>%  
 summarize(mean\_socexp10= mean(socexp10,na.rm=T),  
 median\_socexp10= median(socexp10,na.rm=T),  
 mean\_infl10= mean(infl10,na.rm=T),  
 median\_infl10= median(infl10,na.rm=T))

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| feddummy | legeng | mean\_socexp10 | median\_socexp10 | mean\_infl10 | median\_infl10 |
| Ja | Ja | 0.0920559 | 0.0774453 | 1.023112 | 1.024465 |
| Ja | Nein | 0.0983892 | 0.0949270 | 1.042920 | 1.017862 |
| Nein | Ja | 0.1198066 | 0.1198066 | 1.022365 | 1.019796 |
| Nein | Nein | 0.1547823 | 0.1467336 | 1.072695 | 1.030965 |

# Frage 5

Berechne diese Werte für dünn, mäßig und dicht-besiedeltet Staaten. popdens ist die Grundlage dieser Berechnung. Die Definition für dünn, mäßig, und dicht besiedelt solltet ihr selbst bestimmen.

Ein Möglicher Ansatz ist die Einteilung nach Eckwerten der Besiedlung.

quantile(data\_fed$popdens)

## 0% 25% 50% 75% 100%   
## 2.53922 33.85979 109.84365 177.67052 473.61700

summary(data\_fed$popdens)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2.539 33.860 109.844 130.382 177.671 473.617

Ich definiere hier dünn durch weniger als 25 % der Verteilung, und hoch als höher als 75% der Verteilung. Ich übernehme diese Werte direkt aus dem Befehl *quintile* .

data\_fed<-  
 data\_fed %>%   
 mutate(popdens\_factor = factor(case\_when(  
 popdens < quantile(popdens,0.25) ~ "dünn",  
 popdens >= quantile(popdens,0.25) & popdens < quantile(popdens,0.75) ~"mäßig",  
 popdens >= quantile(popdens,0.75) ~"dicht")),  
 ordered=T)

Jetzt können wir wie gehabt fortfahren.

table\_summary\_5<-  
 data\_fed %>%  
 group\_by(popdens\_factor)%>%  
 summarize(mean\_socexp10= mean(socexp10,na.rm=T),  
 median\_socexp10= median(socexp10,na.rm=T),  
 mean\_infl10= mean(infl10,na.rm=T),  
 median\_infl10= median(infl10,na.rm=T))

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| popdens\_factor | mean\_socexp10 | median\_socexp10 | mean\_infl10 | median\_infl10 |
| dicht | 0.1409430 | 0.1515998 | 1.020092 | 1.018256 |
| dünn | 0.1210850 | 0.0975029 | 1.021244 | 1.020968 |
| mäßig | 0.1369198 | 0.1291446 | 1.100111 | 1.044405 |

# Frage 6

Jetzt können wir uns auch etwas inhaltlicher mit dem Papier auseinandersetzen. Die Frage, wie wir Performanz definieren sollte theoretisch geleitet sein.

Das ist nicht besonders einfach - denn sowohl Performanz als auch die Definitionen der unterschiedlichen Föderalismustypen sind nicht klar definiert. Hier müssen wir Theorie und Daten verbinden.

Eine Möglichkeit ist die Definition anhand der sub-nationalen fiskalischen Autonomie. Aber schon hier merken wir, auch das ist ein Messwert für Dezentralisierung.

data\_fed<-  
 data\_fed %>%   
 mutate(dezentralsierung = factor(case\_when(dezrev1 < 0.06 ~ "Niedrig",  
 dezrev1 >= 0.06 & dezrev1 < 0.24 ~ "Mittel",  
 dezrev1 >= 0.24 ~"Hoch")))  
  
 table\_6<-data\_fed %>%   
 group\_by(feddummy,dezentralsierung) %>%  
 summarize(bip\_kopf= mean(gdpppp94, na.rm=T),  
 bip\_wachstum = mean(growth10, na.rm=T),  
 count= n(),  
 Länder= paste(country,collapse=",")) %>%  
 filter(!is.na(dezentralsierung))

## Warning: Factor `dezentralsierung` contains implicit NA, consider using  
## `forcats::fct\_explicit\_na`

table\_6

## # A tibble: 5 x 6  
## # Groups: feddummy [2]  
## feddummy dezentralsierung bip\_kopf bip\_wachstum count Länder   
## <fct> <fct> <dbl> <dbl> <int> <chr>   
## 1 Ja Hoch 23746. 1.02 4 Canada,Germany,Switzerl~  
## 2 Ja Mittel 17606. 0.997 4 Australia,Austria,Belgi~  
## 3 Nein Hoch 19991. 1.03 3 Denmark,Finland,Sweden   
## 4 Nein Mittel 19319. 1.03 9 Czech Republic,France,H~  
## 5 Nein Niedrig 18737. 1.03 7 Greece,Ireland,Luxembou~

kable(table\_6)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| feddummy | dezentralsierung | bip\_kopf | bip\_wachstum | count | Länder |
| Ja | Hoch | 23745.92 | 1.0165077 | 4 | Canada,Germany,Switzerland,United States |
| Ja | Mittel | 17605.88 | 0.9974353 | 4 | Australia,Austria,Belgium,Mexico |
| Nein | Hoch | 19990.51 | 1.0267987 | 3 | Denmark,Finland,Sweden |
| Nein | Mittel | 19319.02 | 1.0263704 | 9 | Czech Republic,France,Hungary,Iceland,Italy,New Zealand,Norway,Poland,Spain |
| Nein | Niedrig | 18737.17 | 1.0347759 | 7 | Greece,Ireland,Luxembourg,Netherlands,Portugal,Slovakia,United Kingdom |

Grundsätzlicher ist es auch problematisch, dass hier die USA und Deutschland in der gleichen Kategorie sind, obwohl sie als Archetypen des Verbunds - und des Dualen Föderalismus angesehen werden können.

Die Antwort ist also, dass es hier keine richtig gute Antwort innerhalb des Datensatzes gibt.

# Andere Fragen aus dem Seminar

## Faktoren neu Ordnen

Wir können R sagen, dass wir die Faktoren neu ordnen wollen.

data\_fed$popdens\_factor <- factor(data\_fed$popdens\_factor, levels=c("dünn", "mäßig", "dicht"))

summary(data\_fed$popdens\_factor)

## dünn mäßig dicht   
## 8 14 8

## Faktoren aus zwei Variablen kombinieren

Wenn wir eine kategorische Variable kombinieren wollen, können wir weiterhin den Befehl *dplyr::case\_when* nutzen, und die notwendigen Konditionen mit einem “&” verbinden. Wenn wir z.B. unitär und föderal , sowie ein englisches Rechtsssytem abbilden wollen.

data\_fed<-  
 data\_fed %>%   
 mutate(fed\_leg = factor(case\_when(  
 feddummy == "Ja" & legeng == "Ja" ~ "Föderalstaat mit englischem Rechtssystem" ,  
 feddummy == "Ja" & legeng == "Nein" ~ "Föderalstaat ohne englisches Rechtssystem" ,  
 feddummy == "Nein" & legeng == "Ja" ~ "Unitärer Staat mit englischem Rechtssystem" ,  
 feddummy == "Nein" & legeng =="Nein" ~ "Unitärer Staat ohne englisches Rechtssystem")))