

Garagenflohmarkt und Ungleichheit

Karsten Lübke

30.08.2018

1. Forschungsfrage

Ungleichheit ist u.a. in Wissenschaft und Politik ein aktuell viel diskutiertes Thema. Aber wie entsteht sie? Ein einfaches Modell, das sogenannte *Garagenflohmarktmodell* ("Yard Sale Model", (Hayes 2002)) kann dazu dienen ungleiche Vermögensverteilung zu erklären.

Durch einfache Simulationen soll untersucht werden, wie sich z. B. Steuern auf die Entwicklung der Ungleichheit innerhalb des Modells auswirken.¹

2. Garagenflohmarktmodell und Steuern

Garagenflohmarktmodell

Innerhalb einer endlichen, konstanten Population von n Agenten werden nacheinander t Transaktionen zwischen zwei zufällig ausgewählten Händlern ausgeführt. Dabei wird ein Anteil a des Vermögens des ärmeren der beiden Händler ausgetauscht. Beide Händler haben die gleiche Gewinnwahrscheinlichkeit (Nullsummenspiel).

Beispiel:

Es werden zum Zeitpunkt t zufällig Händler A und Händler B ausgewählt. Das Vermögen von Händler A beträgt $x_t^A = 10$ und das von B sei $x_t^B = 50$. Der Anteil des Handelsvolumens sei $a = 10\%$, also $v_t = a \cdot \min(x_t^A, x_t^B) = 0.1 \cdot 10 = 1$. Zufällig gewinnt B, so dass $x_{t+1}^A = 10 - 1 = 9$ und $x_{t+1}^B = 50 + 1 = 51$ beträgt.

Steuern

Steuern können z. B. simuliert werden, so dass alle t_{tax} Transaktionen ein zum Vermögen $x_{t_{tax}}$ proportionaler Steuersatz i erhoben wird und die Summe gleichmäßig auf alle verteilt wird:

$$x_{t_{tax}+1}^j = (1 - i) \cdot x_{t_{tax}}^j + \frac{\sum_{j=1}^n i \cdot x_{t_{tax}}^j}{n}$$

¹Für realistischere Ansätze siehe z. B. (Bertotti and Modanese 2018)

3. Simulation

Vorbereitungen

Pakete laden:

```
# Ggfs. Paket dplyr installieren
# install.packages("dplyr")
library(dplyr)
# Ggfs. Paket tidyr installieren
# install.packages("tidyr")
library(tidyr)
# Ggfs. Paket ggformula installieren
# install.packages("ggformula")
library(ggformula)
# Ggfs. Paket ineq installieren
# install.packages("ineq")
library(ineq)
```

Modellparameter festlegen

Die konkrete Entwicklung der Vermögensverteilung hängt von vielen Parametern ab. Diese werden hier festgelegt.

```
# Anzahl Agenten:
n <- 1000
# Anzahl Transaktionen
trades <- 100000
# Anteil Handelsvolumen
a <- 0.1
# Steuersatz
i <- 0.1
# Fällig nach t_tax Transaktionen
ttax <- 1000
```

Simulation ohne Steuern

Zum Start der Simulation haben alle das gleiche Vermögen, z. B. 100:

```
wealth_oS <- rep(100, n)
```

Vektor, der die Ungleichheit des Vermögens mit Hilfe des Gini-Koeffizienten misst bereitstellen:

```
gini_oS <- numeric(trades)
```

Schleife über die Transaktionen:

```
# Zufallszahlengenerator setzen:
set.seed(1896)

for (j in 1:trades)
{
  # Zufallsauswahl zweier Händler:
  trader <- sample(1:n, 2)
```

```

# Bestimmung Handelsvolumen:
v <- a*min(wealth_oS[trader])

# Der erste Handelspartner gewinnt, der zweite verliert:
wealth_oS[trader[1]] <- wealth_oS[trader[1]] + v
wealth_oS[trader[2]] <- wealth_oS[trader[2]] - v

# Berechnung Gini Koeffizient:
gini_oS[j] <- Gini(wealth_oS)
}

```

Simulation mit Steuern

Zum Start der Simulation haben alle das gleiche Vermögen, z. B. 100:

```
wealth_mS <- rep(100, n)
```

Vektor, der die Ungleichheit des Vermögens mit Hilfe des Gini-Koeffizienten misst bereitstellen:

```
gini_mS <- numeric(trades)
```

Schleife über die Transaktionen:

```

# Zufallszahlengenerator setzen:
set.seed(1896)

for (j in 1:trades)
{
  # Zufallsauswahl zweier Händler:
  trader <- sample(1:n, 2)
  # Bestimmung Handelsvolumen:
  v <- a*min(wealth_mS[trader])

  # Der erste Handelspartner gewinnt, der zweite verliert:
  wealth_mS[trader[1]] <- wealth_mS[trader[1]] + v
  wealth_mS[trader[2]] <- wealth_mS[trader[2]] - v

  # Alle ttax Transaktionen: Steuern fällig.
  # In dem Fall Division Anzahl Iterationen durch ttax ergibt Rest 0.
  if (j%%ttax==0)
  {
    # Vermögenssteuer:
    tax <- i*wealth_mS
    # Steuerabzug:
    wealth_mS <- wealth_mS - tax
    # Steuerausstattung:
    wealth_mS <- wealth_mS + sum(tax)/n
  }

  # Berechnung Gini Koeffizient:
  gini_mS[j] <- Gini(wealth_mS)
}

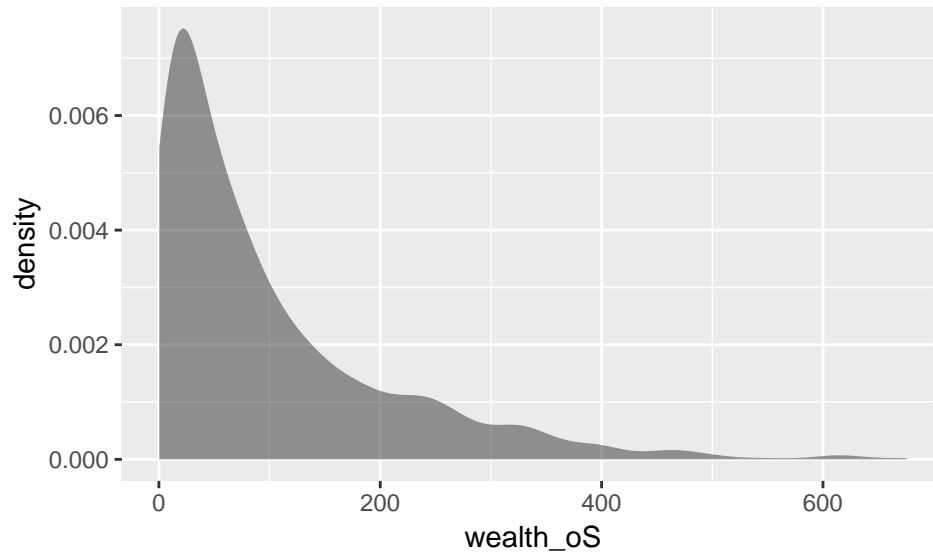
```

4. Ergebnisse

Vermögensverteilung

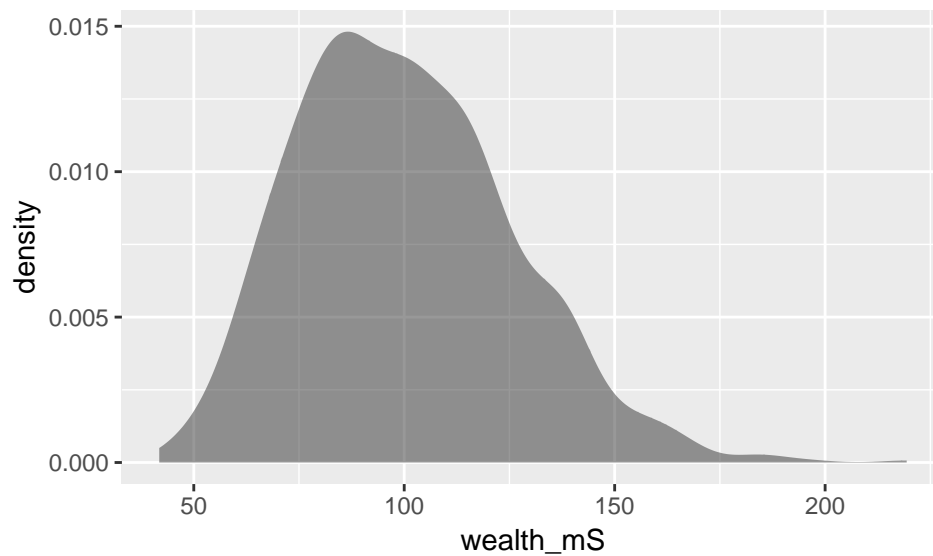
Die Vermögensverteilung am Ende sieht für die Simulation ohne Steuern wie folgt aus:

```
gf_density(~wealth_oS)
```



Während Sie mit Steuern so aussieht:

```
gf_density(~wealth_mS)
```



Während ohne Steuern die Meisten nur noch ein sehr geringes Vermögen haben - nur einige wenige besitzen viel bis sehr viel, ist beim Modell mit Steuern die Verteilung zwar auch rechtsschief, aber nicht so extrem.

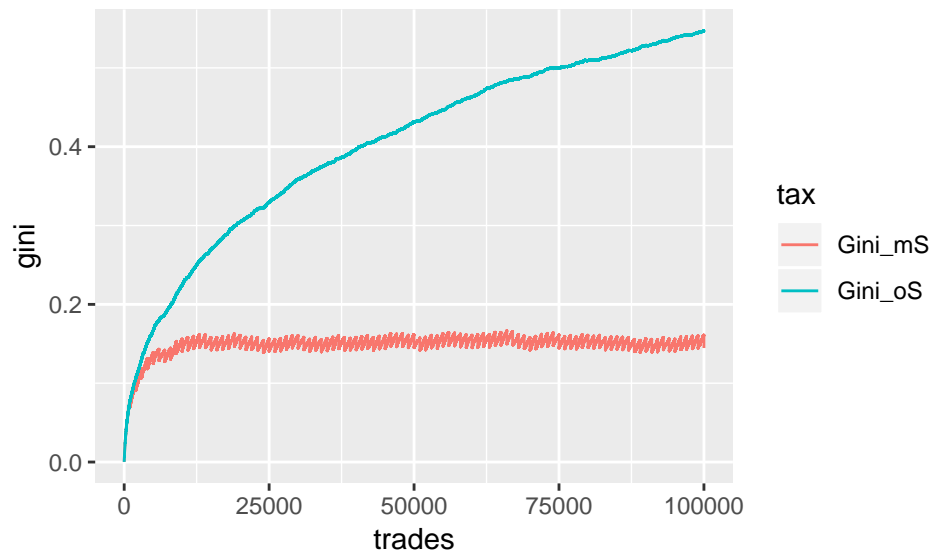
Entwicklung Gini Koeffizient

Datensatz bereitstellen:

```
gini <- data.frame(trades=1:trades, Gini_oS=gini_oS, Gini_mS=gini_mS)
gini <- gini %>%
  gather(tax, gini, -trades)
```

Liniendiagramm:

```
gf_line(gini ~ trades, col=~tax, data = gini)
```



Während der Gini Koeffizient im Fall ohne Steuern immer steigt scheint er in der vorliegenden Simulation mit Steuern nach ca. 15000 Transaktionen eine obere Schranke zu erreichen. Wo diese liegt hängt aber von den gewählten Parametern ab (vgl. (Chorro 2016)).

5. Schlussfolgerungen und Ausblick

Ein stark vereinfachtes Modell (“Garagenflohmarkt”) kann die Entstehung von Ungleichheit auch bei Nullsummenspielen in abgeschlossenen Systemen erklären. Steuern können ein Instrument sein diese abzumildern.

Für mögliche Erweiterungen des Basismodells siehe z. B. (Boghosian 2014).

Interessant ist auch ein Vergleich der theoretischen Modelle mit empirischen Beobachtungen, vgl. z. B. (Brzezinski 2014).

Anhang: Versionshinweise

- Datum erstellt: 2019-06-11
- R Version: 3.6.0

Verwendete Pakete:

- dplyr Version: 0.8.1
- ggformula Version: 0.9.1

- `ineq` Version: 0.2.13
- `tidyr` Version: 0.8.3

Literatur

Bertotti, M. L., and G. Modanese. 2018. “Mathematical Models Describing the Effects of Different Tax Evasion Behaviors.” *Journal of Economic Interaction and Coordination* 13 (2): 351–63. <https://doi.org/10.1007/s11403-016-0185-9>.

Boghosian, Bruce M. 2014. “Kinetics of Wealth and the Pareto Law.” *Physical Review E* 89 (4): 042804.

Brzezinski, Michal. 2014. “Do Wealth Distributions Follow Power Laws? Evidence from ‘Rich Lists’” *Physica A: Statistical Mechanics and Its Applications* 406: 155–62. <https://doi.org/https://doi.org/10.1016/j.physa.2014.03.052>.

Chorro, Christophe. 2016. “A Simple Probabilistic Approach of the Yard-Sale Model.” *Statistics & Probability Letters* 112: 35–40. <https://doi.org/https://doi.org/10.1016/j.spl.2016.01.012>.

Hayes, Brian. 2002. “Computing Science: Follow the Money.” *American Scientist* 90 (5): 400–405.