Eine neoklassische Variante des Malthus Wachstumsmodells

Claudius Gräbner

4/14/2020

Inhaltsverzeichnis

1 Einführung

Dieses Dokument beschreibt eine neoklassische Variante des Malthus Wachstumsmodells. Neben der Einführung in das Modell soll dabei auch die Implementierung in R erkäutert werden. Das in diesem Text eingeführte Modell bildet auch die Grundlage für die Shiny-App, welche die Vorlesung zu Malthus ergänzt.

Die folgenden R-Pakete werden dabei im Text verwendet:

```
library(tidyverse)
library(latex2exp)
```

2 Das Malthus-Wachstumsmodell ohne Kapitalakkumulation.

2.1 Die Bevölkerungsdynamik

Wir nehmen an, dass die Bevölkerung zum Zeitpunkt t gegeben ist durch N_t. Die Veränderung der Bevölkerung über die Zeit hängt von der Geburten- und Sterberate ab. Die Geburtenrate sei konstant und gegeben durch $b = 0.03$. Die Sterberate ist abhängig vom Konsum c_t und sei gegeben durch $d(c_t) = 0.01 - \log(c_t)$. Die Veränderung der Bevölkerung ergibt sich aus der Differenz von Geburten- und Sterberate und sei damit gegeben durch: $g(c_t) = b - d(c_t) = 0.01 - \log(c_t)$, bzw. $G(c_t) = 1 + g(c_t)$ Wir haben also:

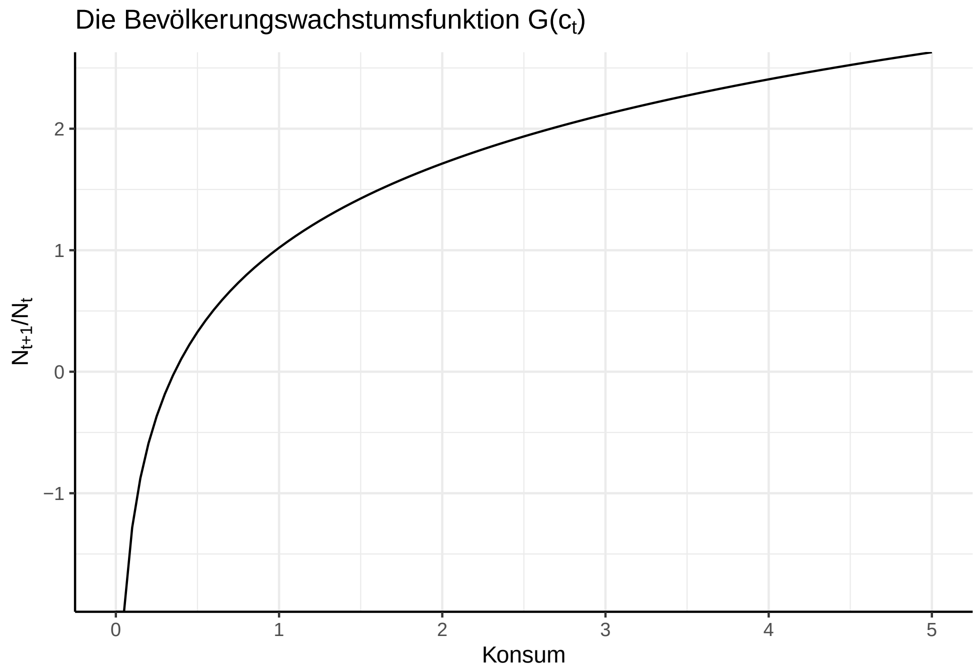
$$N_{t+1} = N_t(1 + g(c_t)) \tag{1}$$

$$N_{t+1} = N_t G(c_t) \tag{2}$$

```
birth_rate <- 0.03
death_rate <- function(consumption){0.01-log(consumption)}
g_c <- function(consumption, birthrate){birthrate - death_rate(consumption)}
G_c <- function(consumption, birthrate){1 + g_c(consumption, birthrate)}
```

Grafisch sieht die Funktion folgendermaßen aus:

```
ggplot(data.frame(x=c(0,5)), aes(x)) +
  stat_function(fun=G_c, args = list(birthrate = birth_rate)) +
  ggtitle(TeX("Die Bevölkerungswachstumsfunktion $G(c_t)$")) +
  xlab("Konsum") + ylab(TeX("$N_{t+1}/N_t$")) +
  scale_y_continuous(expand = expansion()) +
  theme_bw() + theme(panel.border = element_blank(),
                     axis.line = element_line())
```



2.2 Produktion

Wir nehmen eine typische neoklassische Produktionsfunktion mit konstanten Skalenerträgen und abnehmendem Grenzertrag für jeden separaten Produktionsfaktor an. Die Produktionsfaktoren sind Arbeit und Boden, wobei letzterer fix durch $L = L_t \forall t$ und erster durch die Bevölkerung N_t gegeben ist. Entsprechend ist die Produktionsfunktion gegeben durch:

$$Y_t = AL^{\alpha}N_t^{1-\alpha}$$

wobei A einen Paramter für die *totale Faktorproduktivität* (TFT) darstellt. Die TFP können wir als ein allgemeines Maß für technischen Fortschritt interpretieren.

In R:

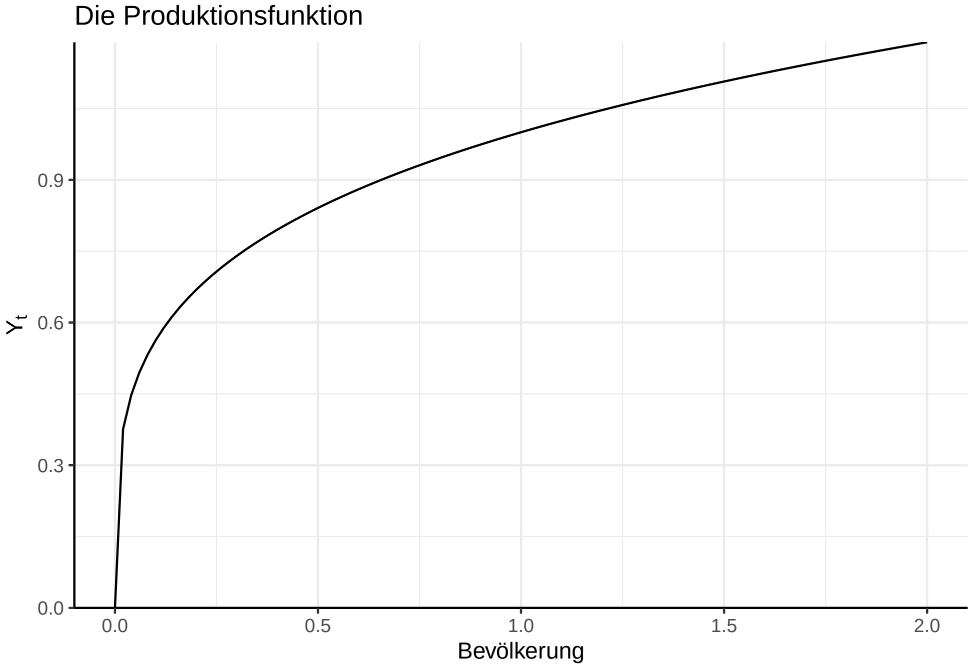
```
L_0 <- 1
N_0 <- 1
tfp <- 1.0
alpha_used <- 0.75

production <- function(total_factor_productivity, land, population, alpha_value){
```

```
  total_factor_productivity * land**alpha_value * population**(1-alpha_value)
}
```

Die Produktionsfunktion sieht dabei folgendermaßen aus:

```
ggplot(data.frame(x=c(0,2)), aes(x)) +
  stat_function(fun=production, args = list(total_factor_productivity = tfp,
                                            land=L_0, alpha_value=alpha_used)) +
  ggtitle("Die Produktionsfunktion") +
  scale_y_continuous(expand = expansion()) +
  xlab("Bevölkerung") + ylab(TeX("$Y_t$")) +
  theme_bw() + theme(panel.border = element_blank(),
                     axis.line = element_line())
```



2.3 Allgemeines Gleichgewicht

2.3.1 Gleichgewichtsbedingungen

Die Modellökonomie besteht aus drei Märkten, die simultan im Gleichgewicht sein müssen: dem Gütermarkt, dem Landmarkt und dem Arbeitsmarkt. Da wir davon ausgehen, dass jede Person ihre gesamte Arbeitskraft (die wir als 1 normieren) anbietet und das angebotene Land mit L fixiert ist, sind diese beiden Gleichgewichtswerte bereits gegeben.

Der Gütermarkt ist im Gleichgewicht wenn der gesamte Output konsumiert wird, also gilt:

$$N_t c_t = Y_t$$

Es fehlen nun noch die Faktorpreise für Land (r_t) und Arbeit (w_t). Diese ergeben sich aus dem Optimierungsproblem der repräsentativen Firma. Erinnern wir uns, dass eine profitmaxiemierende Firma die Produktionsfaktoren zum Preis ihres Grenzertrags nachfragt. Um diese Werte zu bekommen leiten wir die Profitfunktion

$$AL^\alpha N_t^{1-\alpha} - w_t N_t - r_t L_t$$

nach den beiden Produktionsfaktoren ab und setzen sie gleich Null:

$$\frac{\partial AL^{\alpha}N_t^{1-\alpha}-w_tN_t-r_tL_t}{\partial N}=(1-\alpha)AL_t^{\alpha}N_t^{-\alpha}-w_t=(1-\alpha)\frac{Y_t}{N_t}-w_t\overset{!}{=}0$$

sowie

$$\frac{\partial AL^{\alpha}N_t^{1-\alpha}-w_tN_t-r_tL_t}{\partial L}=\alpha AL_t^{\alpha-1}N_t^{1-\alpha}-r_t=\alpha\frac{Y_t}{L_t}-r_t\overset{!}{=}0$$

Daraus ergeben sich dann die Werte für w_t^* und r_t^*:

$$w_t^*=(1-\alpha)\frac{Y_t}{N_t}$$

und

$$r_t^*=\alpha\frac{Y_t}{L_t}$$

Diese Berechnung halten wir auch in Form einer R-Funktion fest:

```
get_w_ss <- function(alpha_values, y_t, n_t){
  (1-alpha_values) * (y_t/n_t)
}

get_r_ss <- function(alpha_values, y_t, l_t){
  alpha_values*(y_t/l_t)
}
```

2.3.2 Gleichgewicht

Da die Bevölkerung im Gleichgewichtspfad konstant ist muss gelten:

$$N_{t+1}=N_t=N^*$$

Da das Bevölkerungswachstum $G(c_t)$ am Ende eine Funktion vom Konsum ist, können wir den Konsum im Gleichgewicht, c^*, über die Bedingung

$$G(c^*)=\frac{N_{t+1}}{N_t}\overset{!}{=}1$$

erhalten. In R verwenden wir dazu die Funktion `uniroot`, mit der wir recht einfach die Nullstellen einer Funktion finden können.

```
test_f <- function(x){
  x - 1
}
uniroot(test_f, interval = c(-5, 5), extendInt = "yes")
```

```
#> $root
#> [1] 1
#>
#> $f.root
#> [1] 0
```

```
#>
#> $iter
#> [1] 1
#>
#> $init.it
#> [1] NA
#>
#> $estim.prec
#> [1] 6
```

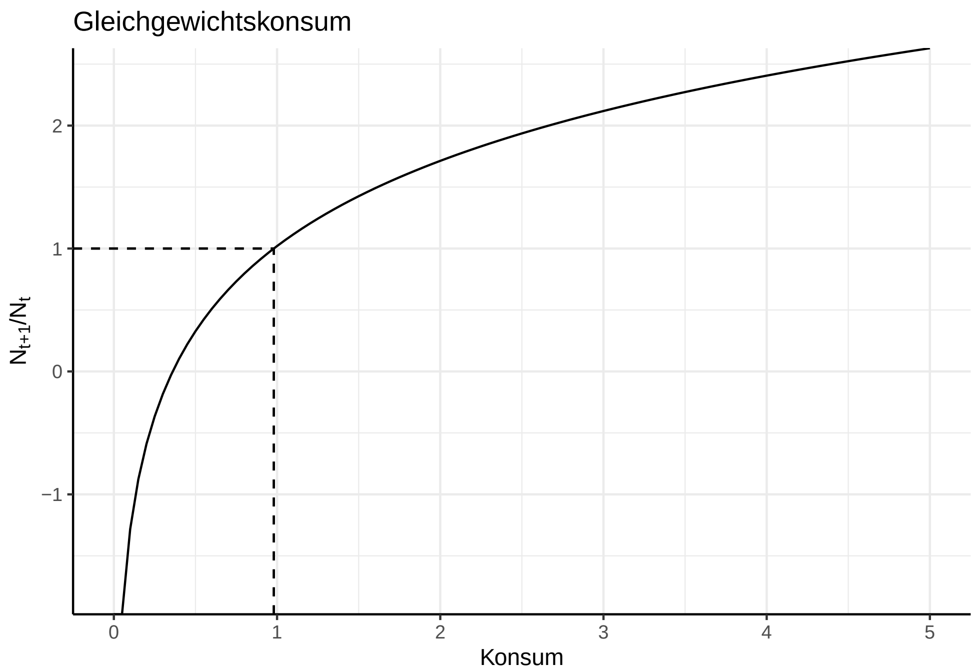
Da wir hier aber den Input suchen für den $G(c)$ den Wert 1 annimmt definieren wir eine Hilfsfunktion:

```
get_c_ss <- function(birth_rate){
  g_c_1 <- function(consumption, birthrate){G_c(consumption, birthrate) - 1}
  uniroot(g_c_1, interval = c(0, 10), birthrate=birth_rate)$root
}
```

Damit hätten wir schon einmal den Konsum im Gleichgewicht:

```
c_ss <- get_c_ss(birth_rate)

ggplot(data.frame(x=c(0,5)), aes(x)) +
  stat_function(fun=G_c, args = list(birthrate = birth_rate)) +
  ggtitle("Gleichgewichtskonsum") +
  scale_y_continuous(expand = expansion()) +
  xlab("Konsum") + ylab(TeX("$N_{t+1}/N_t$")) +
  geom_segment(aes(x = -Inf, y = 1, xend = c_ss, yend = 1),
               linetype="dashed") +
  geom_segment(aes(x = c_ss, y = -Inf, xend = c_ss, yend = 1),
               linetype="dashed") +
  theme_bw() + theme(panel.border = element_blank(),
                     axis.line = element_line())
```



Den Bevölkerungswert im Gleichgewicht erhalten wir durch den Schnittpunkt von $N_t c^*$ und $AL^\alpha N_t^{1-\alpha}$, da wir ja annehmen, dass der Gütermarkt im Gleichgewicht ist.

Damit wir den Schnittpunkt gleich korrekt einzeichnen können berechnen wir ihn wieder mit der

Funktion uniroot:

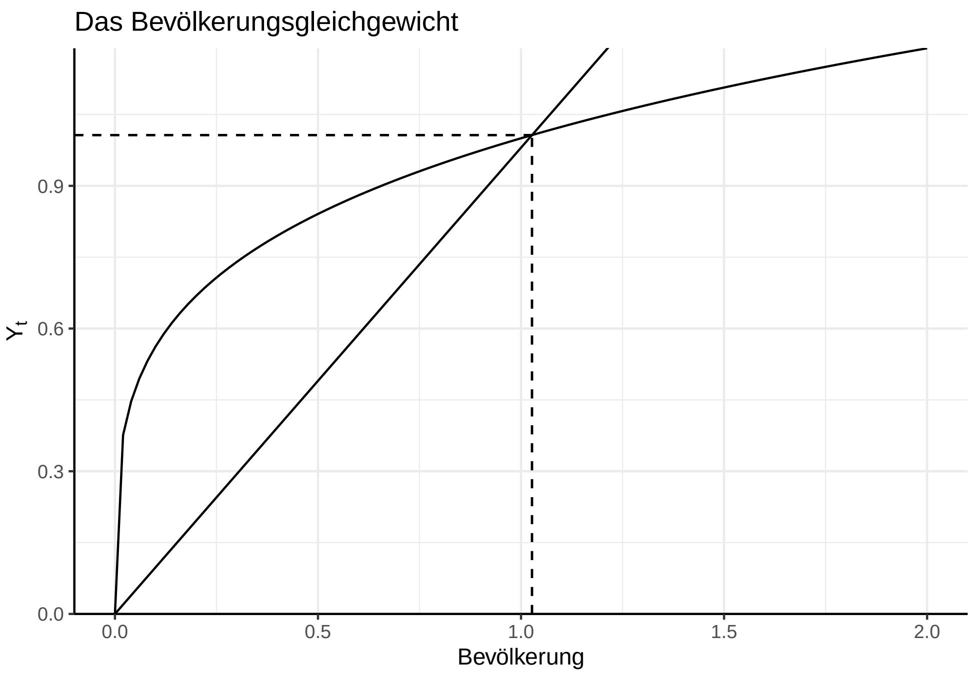
```
interval_check <- seq(0, 2, 0.05)

total_consumption <- function(pop_total, consumption_total){
  pop_total*consumption_total
}

n_ss <- uniroot(function(x) total_consumption(
  x, consumption_total=c_ss)-production(
    population=x,
    total_factor_productivity=tfp,
    land=L_0,
    alpha_value = alpha_used),
  c(0.5, 1.5), extendInt = "yes")$root

y_ss <- production(
    population=n_ss,
    total_factor_productivity=tfp,
    land=L_0,
    alpha_value = alpha_used)
```

```
ggplot(data.frame(x=c(0, 2)), aes(x)) +
  stat_function(fun=production, args = list(total_factor_productivity = tfp,
                                            land=L_0, alpha_value=alpha_used)) +
  ggtitle("Das Bevölkerungsgleichgewicht") +
  scale_y_continuous(expand = expansion()) +
  xlab("Bevölkerung") + ylab(TeX("$Y_t$")) +
  geom_segment(aes(x = -Inf, y = y_ss, xend = n_ss, yend = y_ss),
               linetype="dashed") +
  geom_segment(aes(x = n_ss, y = -Inf, xend = n_ss, yend = y_ss),
               linetype="dashed") +
  geom_abline(intercept = 0, slope = c_ss) +
  theme_bw() + theme(panel.border = element_blank(),
                     axis.line = element_line())
```



Nun können wir die noch fehlenden Faktorpreise berechnen:

```
w_ss <- get_w_ss(alpha_used, y_ss, n_ss)
w_ss
```

```
#> [1] 0.2450503
```

```
r_ss <- get_r_ss(alpha_used, y_ss, L_0)
r_ss
```

```
#> [1] 0.7550161
```

2.4 Statische Gleichgewichtsanalye

Der Übersicht halber sammeln wir hier noch einmal alle relevanten Parameter im Modell:

```
N_0 <- 1
L_0 <- 1
birth_rate <- 0.03
death_rate <- function(consumption){0.01-log(consumption)}
g_c <- function(consumption, birthrate){birthrate - death_rate(consumption)}
G_c <- function(consumption, birthrate){1 + g_c(consumption, birthrate)}
tfp <- 1.0
alpha_used <- 0.75
```

Die folgenden Fragestellungen können Sie nun selbst mit der App beantworten.

2.4.1 Effekt eines erhöhten Ausgangsniveaus der Bevölkerung

2.4.2 Effekt von erhöhter Sterblichkeit

2.4.3 Effekt von technologischer Innovation