

7. ggrepel

Nicht-ueberlappende Textlabels fuer Streudiagramme

Dr. Paul Schmidt

```
for (pkg in c("tidyverse", "gapminder", "ggrepel")) {
  if (!require(pkg, character.only = TRUE)) install.packages(pkg)
}
```

Streudiagramme mit vielen Datenpunkten profitieren oft von Textlabels, die einzelne Beobachtungen identifizieren - Laendernamen, Gen-IDs oder Probenbezeichnungen. Der naheliegendste Ansatz ist `geom_text()`, aber sobald die Anzahl der beschrifteten Punkte ueber eine Handvoll hinausgeht, ueberlappen sich die Labels gegenseitig und verdecken die Datenpunkte, sodass der Plot unleserlich wird.

Das {ggrepel}-Paket loest dieses Problem mit zwei Drop-in-Ersetzungen:

`geom_text_repel()` und `geom_label_repel()`. Beide verwenden einen iterativen

Algorithmus, der Labels voneinander und von den Datenpunkten wegschiebt und duenne Linien-Segmente zeichnet, die jedes Label mit seinem Punkt verbinden. Da die Funktionen dieselbe Aesthetik-Schnittstelle wie `geom_text()` und `geom_label()` haben, reicht es, den Funktionsnamen auszutauschen.

In diesem Kapitel verwendet man den gapminder-Datensatz fuer das Jahr 2007 und stellt das BIP pro Kopf gegen die Lebenserwartung fuer alle 142 Laender dar - ein dichtes Streudiagramm, bei dem die automatische Label-Platzierung einen echten Unterschied macht.

Setup

Man filtert die gapminder-Daten auf das Jahr 2007 und behaelt alle Laender. Da das BIP pro Kopf mehrere Groessenordnungen umfasst (von wenigen hundert bis ueber 40.000 USD), wird durchgehend eine logarithmische x-Achse verwendet.

```
dat <- gapminder::gapminder %>%
  filter(year == 2007)

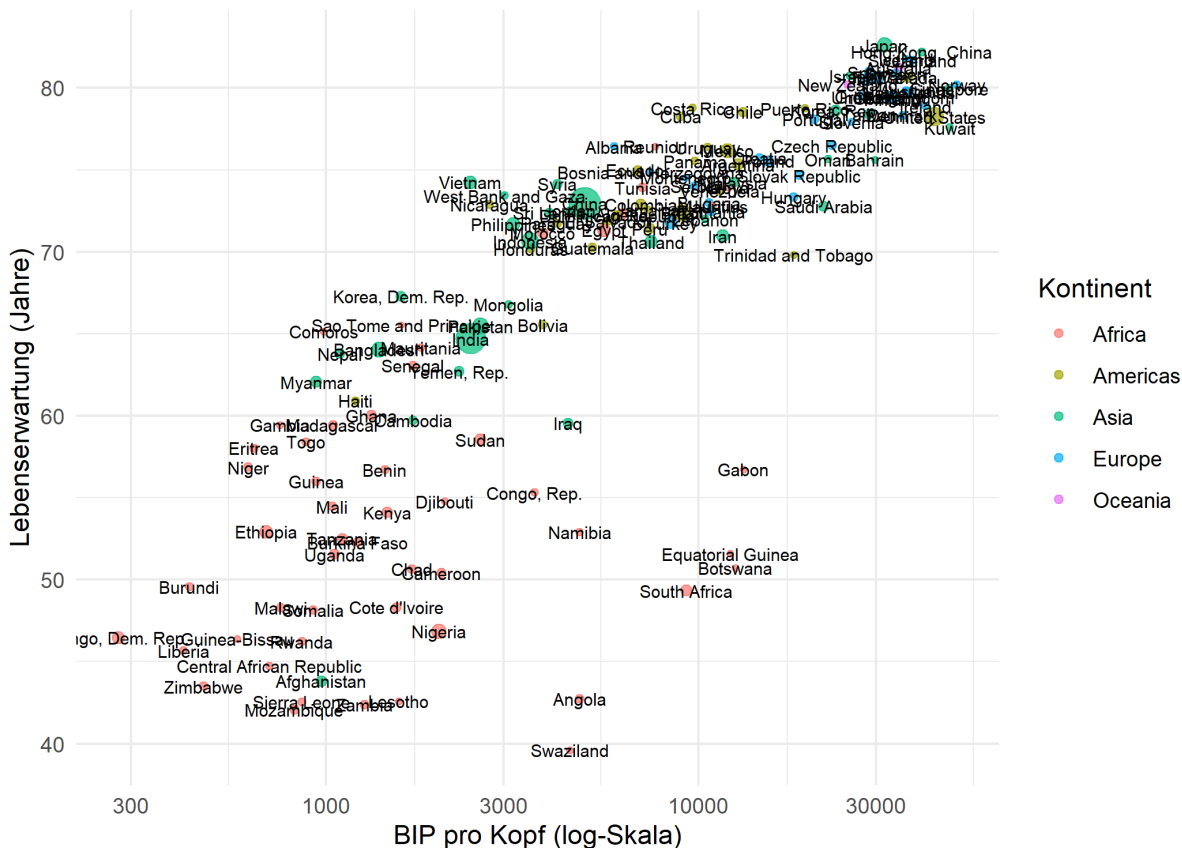
dat
```

```
# A tibble: 142 × 6
  country    continent  year lifeExp      pop gdpPercap
  <fct>      <fct>    <int>  <dbl>    <int>    <dbl>
1 Afghanistan Asia      2007   43.8  31889923    975.
2 Albania    Europe    2007   76.4   3600523   5937.
3 Algeria    Africa    2007   72.3  33333216   6223.
4 Angola     Africa    2007   42.7  12420476   4797.
5 Argentina  Americas  2007   75.3  40301927  12779.
6 Australia  Oceania   2007   81.2  20434176  34435.
7 Austria    Europe    2007   79.8   8199783  36126.
8 Bahrain    Asia      2007   75.6   708573   29796.
9 Bangladesh Asia      2007   64.1 150448339   1391.
10 Belgium   Europe    2007   79.4  10392226  33693.
# i 132 more rows
```

Um das Labeling-Problem zu veranschaulichen, erstellen wir hier ein Streudiagramm mit

`geom_text()` - jeder Ländername wird direkt an seinem Datenpunkt platziert:

```
ggplot(dat, aes(x = gdpPercap, y = lifeExp)) +
  geom_point(aes(size = pop, color = continent), alpha = 0.7) +
  geom_text(aes(label = country), size = 2.5) +
  scale_x_log10() +
  scale_size_continuous(guide = "none") +
  labs(
    x = "BIP pro Kopf (log-Skala)",
    y = "Lebenserwartung (Jahre)",
    color = "Kontinent"
  ) +
  theme_minimal()
```



Das Ergebnis ist ein unleserliches Durcheinander: Labels ueberlappen sich gegenseitig und verdecken die Punkte. Genau zur Loesung dieses Problems wurde `ggrepel` entwickelt.

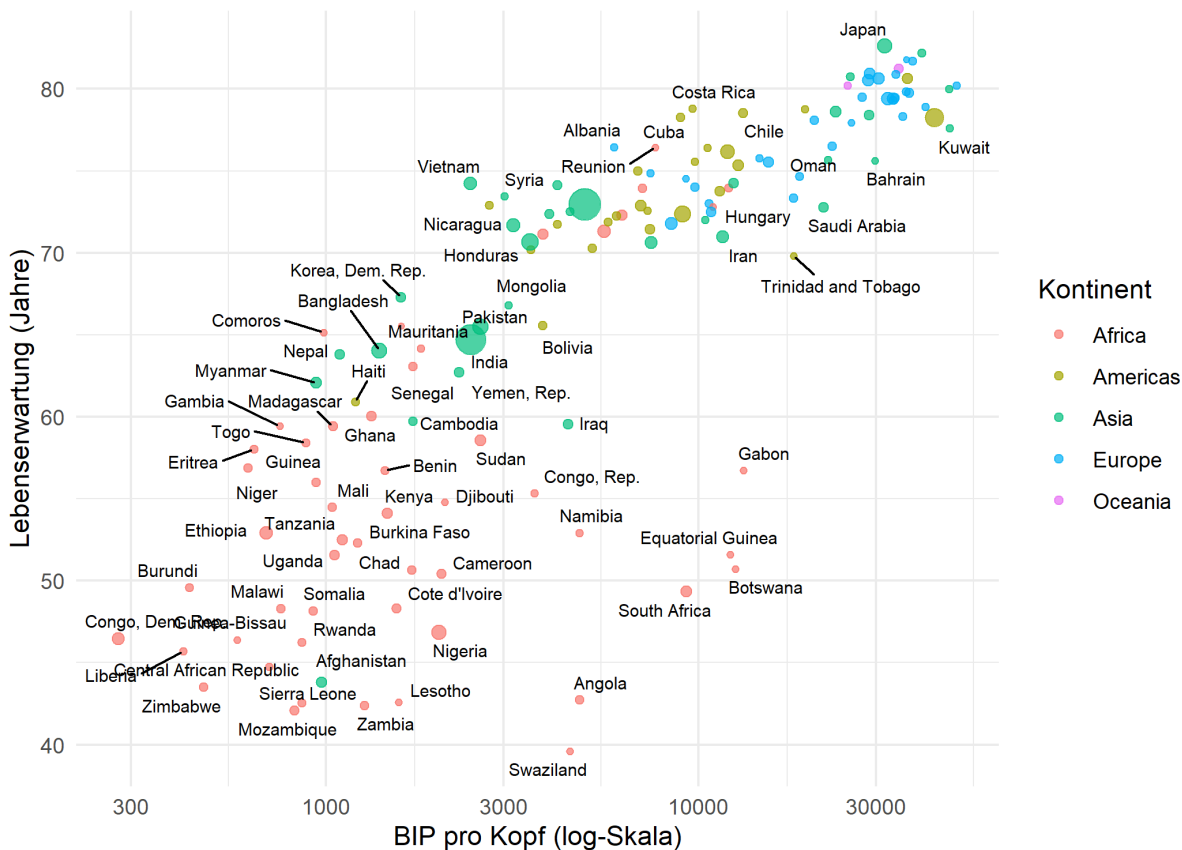
geom_text_repel

Man ersetzt einfach `geom_text()` durch `geom_text_repel()`. Die Funktion positioniert Labels automatisch so um, dass sie sich nicht ueberlappen, und zeichnet ein kleines Liniensegment von jedem Label zu seinem Datenpunkt:

```
ggplot(dat, aes(x = gdpPercap, y = lifeExp)) +
  geom_point(aes(size = pop, color = continent), alpha = 0.7) +
  geom_text_repel(aes(label = country), size = 2.5) +
  scale_x_log10() +
  scale_size_continuous(guide = "none") +
  labs(
    x = "BIP pro Kopf (log-Skala)",
    y = "Lebenserwartung (Jahre)",
    color = "Kontinent"
  )
```

```
) +  
theme_minimal()
```

Warning: ggrepel: 68 unlabeled data points (too many overlaps). Consider increasing max.overlaps



Standardmaessig unterdrueckt `geom_text_repel()` Labels, wenn mehr als 10 in derselben Region ueberlappen wuerden (gesteuert durch `max.overlaps`). Mehrere Parameter helfen bei der Feinabstimmung der Platzierung:

- `max.overlaps` - Maximale Anzahl ueberlappender Labels, bevor ein Label ausgeblendet wird. Der Standard ist 10; man setzt `Inf`, um alle Labels unabhaengig von Ueberlappungen anzuzeigen.
- `box.padding` - Zusaetzlicher Abstand um die Bounding Box jedes Labels (Standard 0.25 Zeilen). Erhoehen fuer mehr Abstand zwischen Labels.
- `point.padding` - Mindestabstand zwischen einem Label und seinem Datenpunkt (Standard 0).
- `min.segment.length` - Segmente kuerzer als dieser Wert werden ausgeblendet (Standard 0.5 Zeilen). Auf 0 setzen, um Segmente immer anzuzeigen.
- `seed` - Der Repulsion-Algorithmus hat eine stochastische Komponente. Durch Setzen eines Seeds erhaelt man bei jedem Rendern dieselben Label-Positionen.

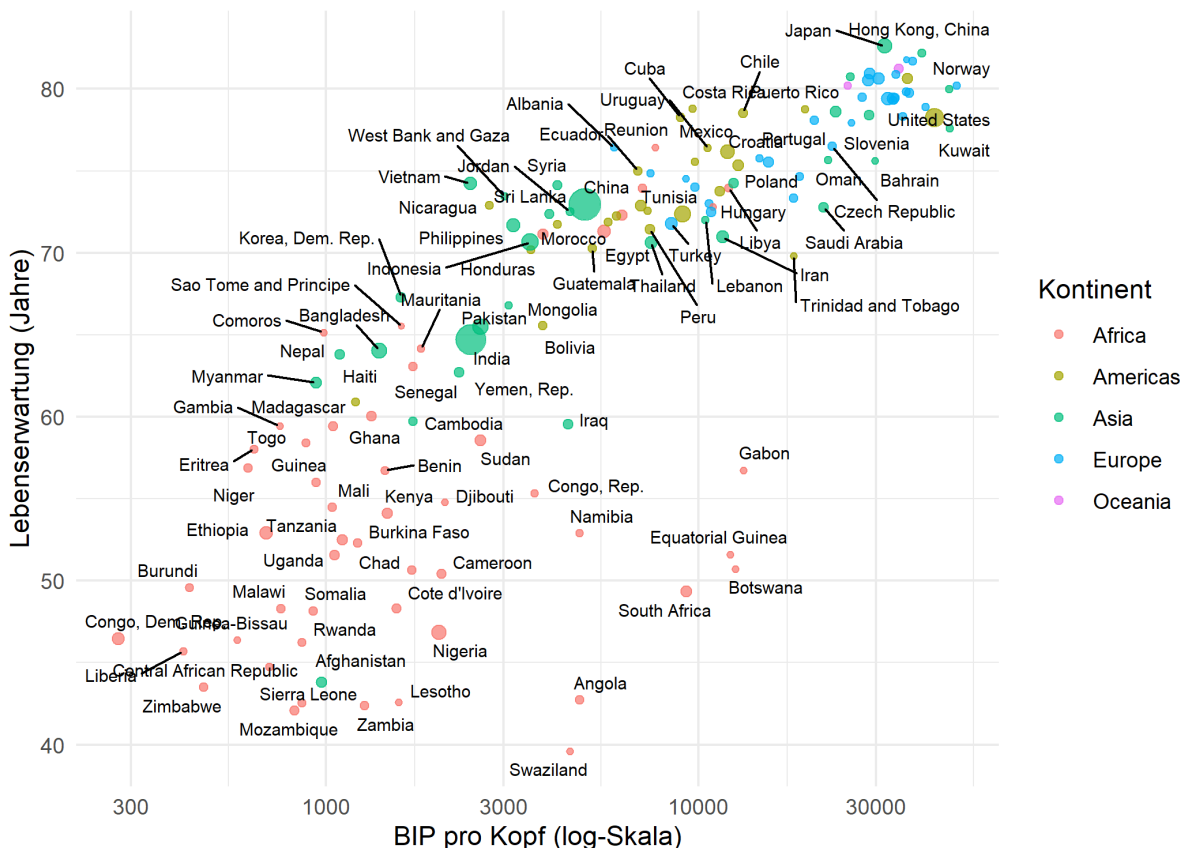
```
ggplot(dat, aes(x = gdpPercap, y = lifeExp)) +  
  geom_point(aes(size = pop, color = continent), alpha = 0.7) +  
  geom_text_repel(  
    aes(label = country),  
    size = 2.5,
```

```

max.overlaps = 20,
seed = 42
) +
scale_x_log10() +
scale_size_continuous(guide = "none") +
labs(
  x = "BIP pro Kopf (log-Skala)",
  y = "Lebenserwartung (Jahre)",
  color = "Kontinent"
) +
theme_minimal()

```

Warning: ggrepel: 40 unlabeled data points (too many overlaps). Consider increasing max.overlaps



geom_label_repel

Während `geom_text_repel()` einfachen Text zeichnet, platziert `geom_label_repel()` den Text in einem gefüllten Rechteck - nützlich, wenn Labels sich von einem unruhigen Hintergrund abheben müssen. Die Verbindungssegmente lassen sich unabhängig gestalten:

```

dat_small <- dat %>%
  filter(continent == "Europe")

ggplot(dat_small, aes(x = gdpPercap, y = lifeExp)) +
  geom_point(aes(size = pop), color = "#201E50", alpha = 0.7) +
  geom_label_repel(
    aes(label = country),
    size = 2.5,
    fill = "white",
    label.size = 0.2,

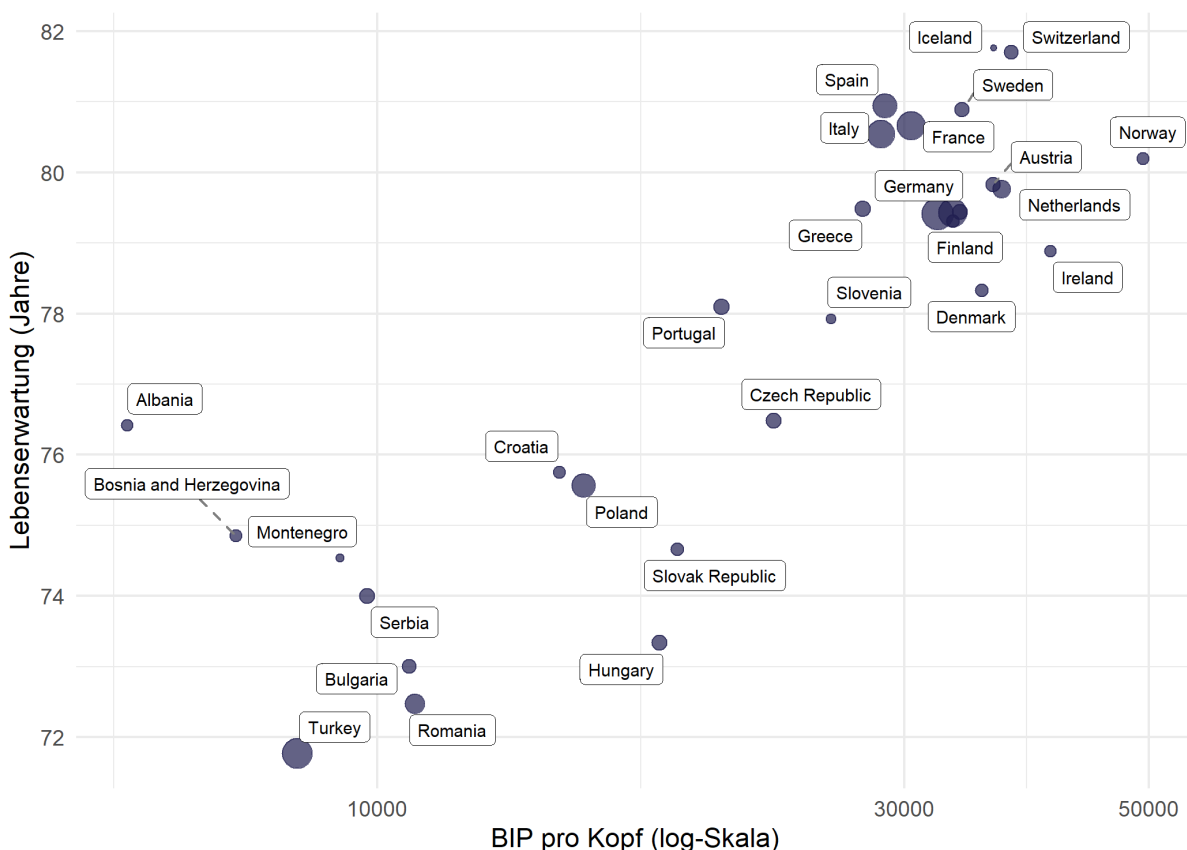
```

```

segment.color = "grey50",
segment.linetype = "dashed",
seed = 42
) +
scale_x_log10() +
scale_size_continuous(guide = "none") +
labs(
  x = "BIP pro Kopf (log-Skala)",
  y = "Lebenserwartung (Jahre)"
) +
theme_minimal()

```

Warning: ggrepel: 2 unlabeled data points (too many overlaps). Consider increasing max.overlaps



Da das Labeln aller 142 Laender auf einmal selbst mit Repulsion ueberfordernd wirkt, ist eine Filterung auf einen einzelnen Kontinent (hier Europa mit 30 Laendern) fuer

`geom_label_repel()` deutlich effektiver.

Selektives Labeling

In vielen Situationen ist es weder noetig noch wuensenswert, jeden Punkt zu beschriften. Eine gaengige Strategie besteht darin, nur die interessantesten Beobachtungen zu beschriften - zum Beispiel die Laender mit der hoechsten Lebenserwartung - und den Rest als unbeschriftete Punkte zu belassen. So bleibt der Plot uebersichtlich und hebt gleichzeitig die wichtigsten Daten hervor.

Der Ansatz ist einfach: Man erstellt eine Hilfsspalte, die den Laendernamen fuer Punkte enthaelt, die beschriftet werden sollen, und `NA` fuer alle anderen. `geom_text_repel()`

ueberspringt `NA`-Werte bei `na.rm = TRUE`:

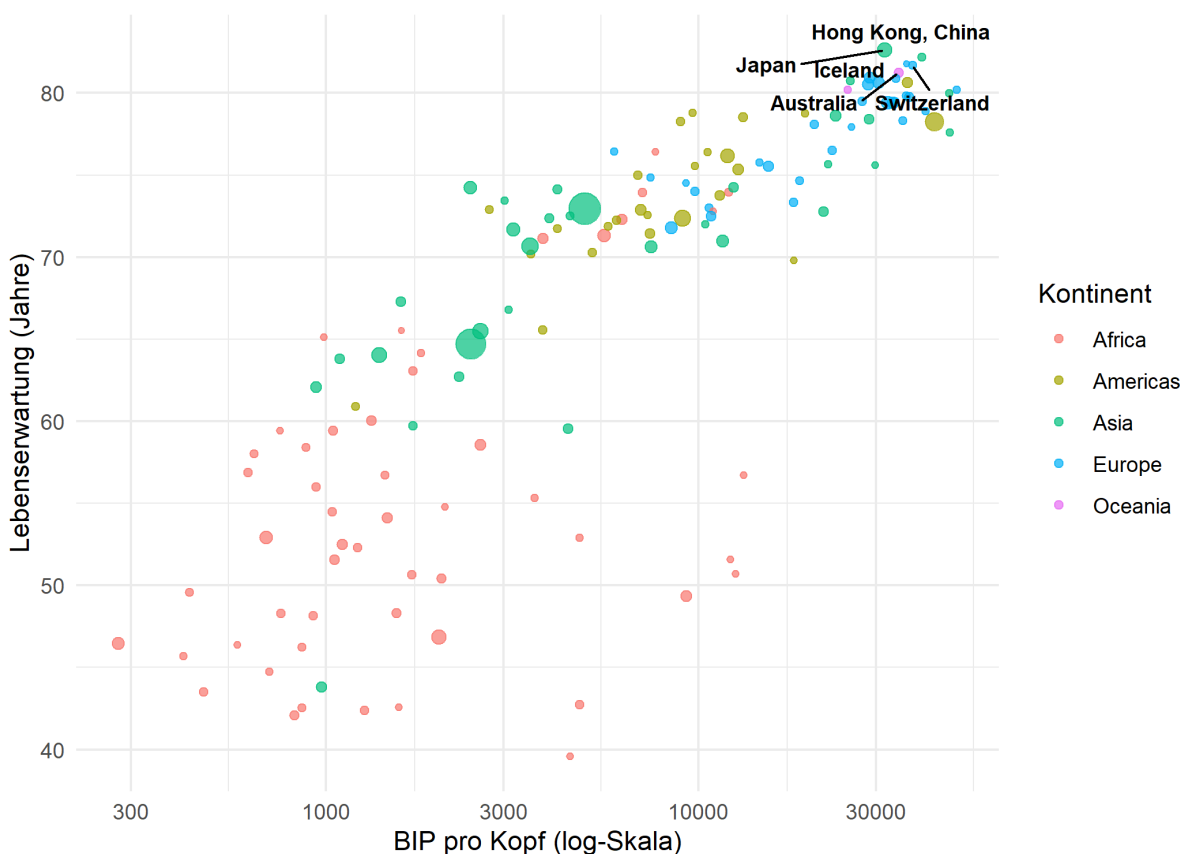
```

top5 <- dat %>%
  slice_max(lifeExp, n = 5) %>%
  pull(country)

dat_labeled <- dat %>%
  mutate(label_col = if_else(country %in% top5, as.character(country),
    NA_character_))

ggplot(dat_labeled, aes(x = gdpPercap, y = lifeExp)) +
  geom_point(aes(size = pop, color = continent), alpha = 0.7) +
  geom_text_repel(
    aes(label = label_col),
    size = 3,
    fontface = "bold",
    seed = 42,
    na.rm = TRUE
  ) +
  scale_x_log10() +
  scale_size_continuous(guide = "none") +
  labs(
    x = "BIP pro Kopf (log-Skala)",
    y = "Lebenserwartung (Jahre)",
    color = "Kontinent"
  ) +
  theme_minimal()

```



Diese Technik skaliert gut: Man kann Ausreisser, Punkte ueber einem Schwellenwert oder jede durch eine logische Bedingung definierte Teilmenge beschriften - alles ohne den zugrundeliegenden Datensatz zu veraendern.

Seitenausgerichtete Labels

Ein besonders aufgeraemter Labeling-Stil platziert alle Labels auf einer Seite des Plots, vertikal ausgerichtet. Das vermeidet das verstreute Erscheinungsbild frei abgestossener

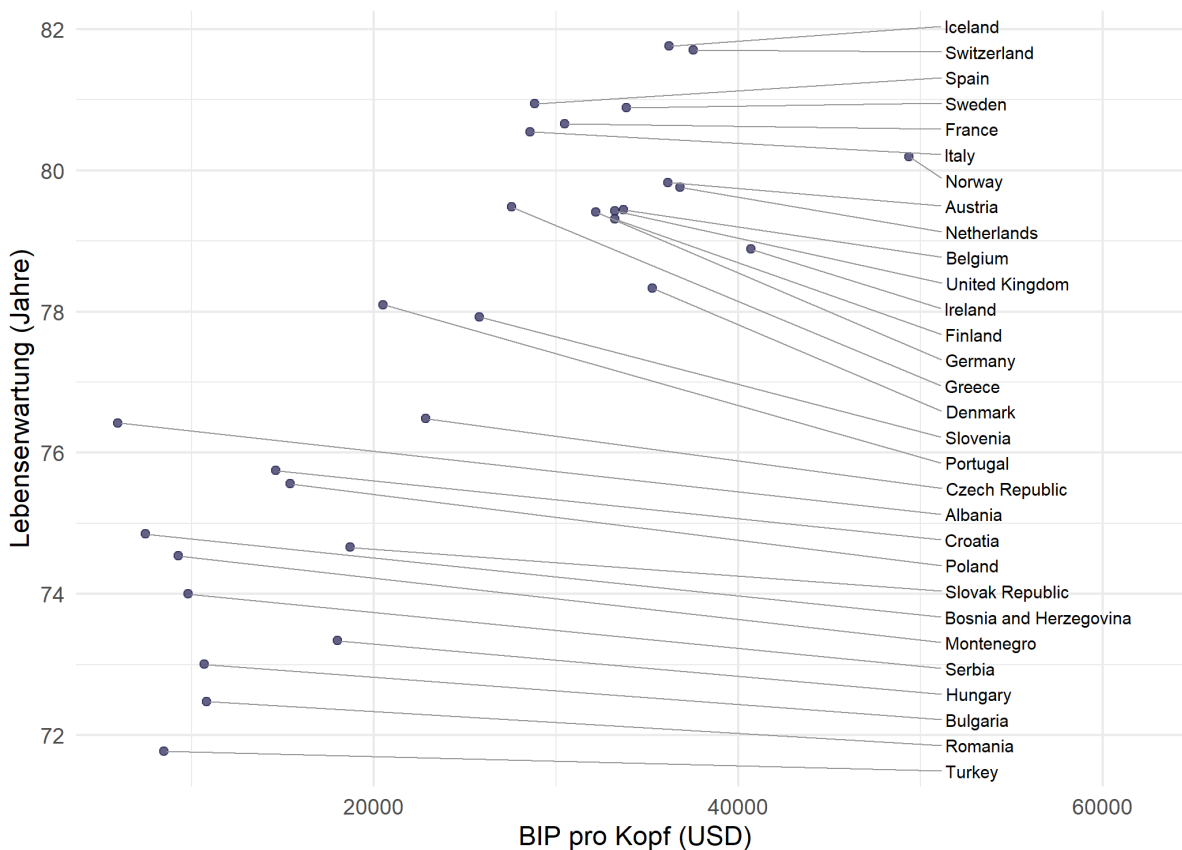
Labels und funktioniert gut, wenn die Anzahl der Punkte ueberschaubar ist. Die Schlüsselparameter sind:

- `direction = "y"` - beschraenkt die Label-Bewegung auf die vertikale Achse, sodass Labels auf derselben x-Position bleiben
- `hjust = 0` - richtet den Labeltext linksbueendig aus
- `nudge_x` - verschiebt alle Labels um einen festen Abstand nach rechts von ihren Datenpunkten
- `expand` auf der x-Achse - schafft genuegend Platz auf der rechten Seite fuer die Labels

```
dat_europe <- dat %>%
  filter(continent == "Europe")

max_gdp <- max(dat_europe$gdpPercap)

ggplot(dat_europe, aes(x = gdpPercap, y = lifeExp)) +
  geom_point(color = "#201E50", alpha = 0.7) +
  geom_text_repel(
    aes(label = country),
    size = 2.5,
    direction = "y",
    hjust = 0,
    nudge_x = max_gdp - dat_europe$gdpPercap + 2000,
    segment.size = 0.3,
    segment.color = "grey60",
    seed = 42
  ) +
  scale_x_continuous(
    name = "BIP pro Kopf (USD)",
    expand = expansion(mult = c(0.05, 0.3))
  ) +
  labs(y = "Lebenserwartung (Jahre)") +
  theme_minimal()
```



Der Trick ist, `nudge_x` pro Punkt als `max(x) - x + offset` zu berechnen, sodass jedes Label unabhaengig von der Position seines Datenpunkts an derselben x-Position landet. In Kombination mit `direction = "y"` breiten sich die Labels nur vertikal von diesem Ankerpunkt aus und bilden eine ordentliche Spalte, die durch gerade horizontale Segmente mit ihren Punkten verbunden ist.

i Performance

Bei Hunderten von Labels kann der Repulsion-Algorithmus langsam werden. Zwei Parameter helfen bei der Steuerung der Rechenzeit: `max.overlaps` begrenzt, wie viele ueberlappende Labels versucht werden (der Rest wird ausgeblendet), und `max.time` setzt ein Zeitlimit fuer den Algorithmus in Sekunden (Standard 0.5). Fuer sehr dichte Plots ist selektives Labeling wie oben gezeigt meist der bessere Ansatz.

Bibliography
