Stichproben aus der Posteriori-Verteilung ziehen

QM2, Thema 3

AWM, HS Ansbach

Gliederung

1. Mit Stichproben die Post-Verteilung zusammenfassen
2. Mit Stichproben neue Beobachtungen simulieren
3. Hinweise
4. Literatur

Mit Stichproben die Post-Verteilung zusammenfassen

Zur Erinnerung: Gitterwerte in R berechnen

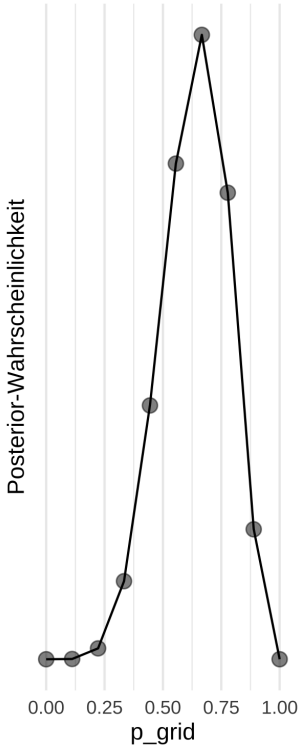
```
n <- 10
n_success <- 6
n_trials  <- 9

d <-
  tibble(p_grid = seq(from = 0, to = 1, length.out = n),
         prior  = 1) %>%
  mutate(likelihood = dbinom(n_success,
                             size = n_trials,
                             prob = p_grid)) %>%
  mutate(unstand_post = (likelihood * prior),
         post = unstand_post / sum(unstand_post))
```

```
## Rows: 10
## Columns: 5
## $ p_grid       <dbl> 0.00, 0.11, 0.22, 0.33, 0.44, 0.56, 0.67, 0.78, 0.89, 1.00
## $ prior        <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
## $ likelihood   <dbl> 0.00, 0.00, 0.00, 0.03, 0.11, 0.22, 0.27, 0.20, 0.06, 0.00
## $ unstand_post <dbl> 0.00, 0.00, 0.00, 0.03, 0.11, 0.22, 0.27, 0.20, 0.06, 0.00
## $ post         <dbl> 0.00, 0.00, 0.01, 0.04, 0.12, 0.24, 0.30, 0.23, 0.06, 0.00
```

Zur Erinnerung, die Gittermethode

Die Gittermethode ist ein Weg, die Posteriori-Verteilung zu berechnen. Die Posteriori-Verteilung birgt viele nützliche Informationen.

Modell: $W = 6$ Wasser, $N = 9$ Würfen und $k = 10$ Gitterwerten.



Die ersten paar Zeilen (von 10) aus der Tabelle d:

Tabelle *d* mit Daten zur Posteriori-Verteilung

p_grid	prior	likelihood	unstand_post	post
0	1	0	0	0
1×10^{-1}	1	1×10^{-4}	1×10^{-4}	1×10^{-4}
2×10^{-1}	1	5×10^{-3}	5×10^{-3}	5×10^{-3}
3×10^{-1}	1	3×10^{-2}	3×10^{-2}	4×10^{-2}
4×10^{-1}	1	1×10^{-1}	1×10^{-1}	1×10^{-1}
6×10^{-1}	1	2×10^{-1}	2×10^{-1}	2×10^{-1}

Befragen wir die Posteriori-Verteilung

Beispiele für Fragen an die Post-Verteilung*:

- Mit welcher Wahrscheinlichkeit liegt der Parameter unter einem bestimmten Wert?
- Mit welcher Wahrscheinlichkeit liegt der Parameter zwischen zwei bestimmten Werten?
- Mit 5% Wahrscheinlichkeit liegt der Parameterwert nicht unter welchem Wert?
- Welcher Parameterwert hat die höchste Wahrscheinlichkeit?
- Wie ungewiss ist das Modell über die Parameterwerte?

Solche Fragen kann man in drei Gruppen aufteilen:

1. Fragen zu Bereichen von Parametern
2. Fragen zu Bereichen von Wahrscheinlichkeitsmassen
3. Fragen zu Punktschätzern von Parametern

*Post-Verteilung: Posteriori-Verteilung

Häufigkeiten sind einfacher als Wahrscheinlichkeiten

Tabelle mit Stichprobendaten aus der Posteriori-Verteilung (Tabelle `d`):

```
samples <-
  d %>%  # nimmt die Tabelle mit Posteriori-Daten,
  slice_sample(  # Ziehe daraus eine Stichprobe,
    n = 1e4,  # mit insgesamt n=10000 Elementen,
    weight_by = post,  # Gewichte nach Spalte mit Post-Wskt.,
    replace = T)  # Ziehe mit Zurücklegen
```

Die Wahrscheinlichkeit, einen Parameterwert aus Tabelle `d` zu ziehen, ist proportional zur Posteriori-Wahrscheinlichkeit (`post`) dieses Werts. Ziehen mit Zurücklegen hält die Wahrscheinlichkeiten während des Ziehens konstant.

Stichprobendaten aus der Post-Verteilung

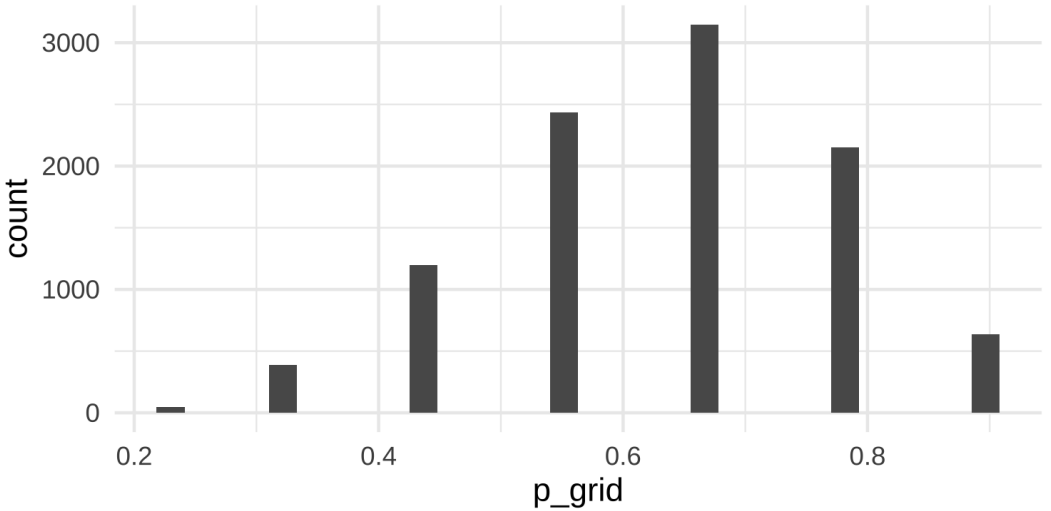
Nur die ersten Zeilen abgebildet

p_grid	prior	likelihood	unstand_post	post
0.889	1	0.057	6×10^{-2}	0.063
0.444	1	0.111	1×10^{-1}	0.123
0.556	1	0.217	2×10^{-1}	0.241

Visualierung von `samples`

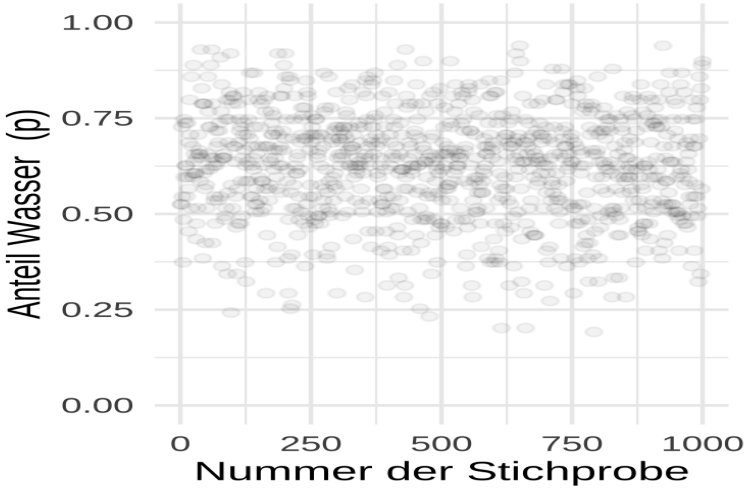
Die ersten 100 gesampelten Gitterwerte (`p_grid`):

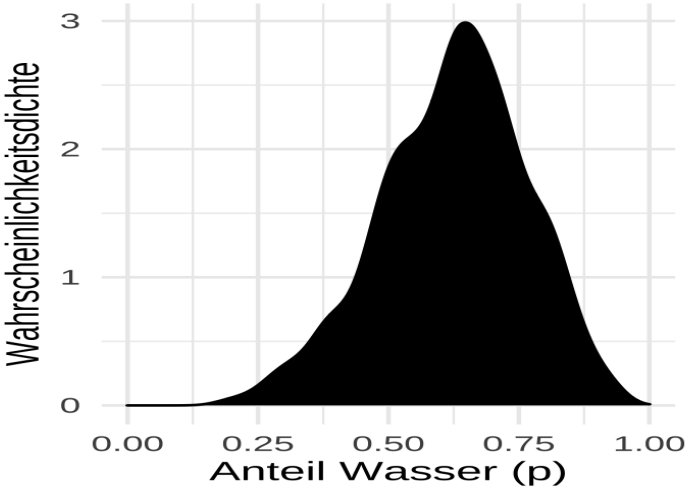
```
##   [1] 0.89 0.44 0.56 0.67 0.56 0.67 0.67 0.44 0.78 0.67 0.67 0.56 0.78 0.56 0.67
##  [16] 0.56 0.56 0.67 0.78 0.44 0.44 0.67 0.67 0.78 0.67 0.67 0.67 0.67 0.67 0.56
##  [31] 0.56 0.56 0.78 0.44 0.67 0.33 0.56 0.67 0.56 0.33 0.56 0.67 0.78 0.67 0.22
##  [46] 0.67 0.67 0.78 0.67 0.89 0.33 0.56 0.56 0.78 0.78 0.89 0.44 0.78 0.78 0.78
##  [61] 0.78 0.67 0.78 0.67 0.56 0.67 0.78 0.44 0.89 0.56 0.56 0.78 0.78 0.56 0.78
##  [76] 0.78 0.56 0.56 0.56 0.67 0.56 0.56 0.33 0.56 0.89 0.44 0.44 0.78 0.67 0.56
##  [91] 0.67 0.67 0.56 0.56 0.67 0.44 0.56 0.67 0.78 0.56
```



Visualisierung der Stichprobendaten mit $k = 100$ Gitterwerten

Datensatz `samples`, $n = 10^3$, $k = 100$ Gitterwerte, basierend auf dem Modell oben.



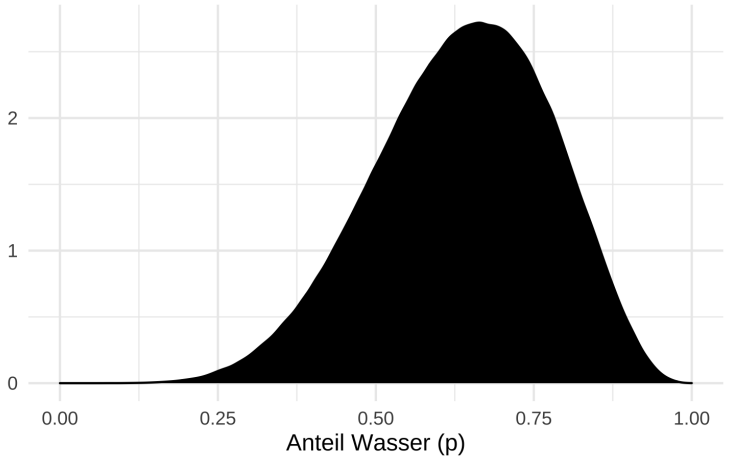


Die Stichprobendaten nähern sich der Posteriori-Verteilung an.

Mehr Stichproben und mehr Gitterwerte glätten die Verteilung

$n = 10^6$ Stichproben bei $k = 100$ Gitterwerten aus der Posteriori-Verteilung

```
d_k100 %>%
  slice_sample(n = 1e6, weight_by = post, replace = T) %>%
  ggplot(aes(x = p_grid)) +
  geom_density(fill = "black") +
  scale_x_continuous("Anteil Wasser (p)", limits = c(0, 1)) +
  labs(y = "")
```



Fragen zu Bereichen von Parametern 1

Wie groß ist die Wahrscheinlichkeit, dass der Wasseranteil unter 50% liegt?

Aus der Posteriori-Verteilung mit der Gridmethode:

```
d %>%
  filter(p_grid < .5) %>%
  summarise(sum = sum(post))
```

```
## # A tibble: 1 × 1
##     sum
##   <dbl>
## 1 0.167
```

Aus den Stichproben der Posteriori-Verteilung:

```
samples %>%
  filter(p_grid < .5) %>%
  summarise(sum = n() / 1e4)
```

```
## # A tibble: 1 × 1
##     sum
##   <dbl>
## 1 0.163
```

Einfach wie 🍰 essen.

Die Gridmethode funktioniert bei großen Modellen nicht gut (im Gegensatz zur quadratischen Approximation `quap`). Daher werden wir ab jetzt mit den Stichproben arbeiten, weil das für `quap` auch funktioniert. Das ist außerdem einfacher.

Fragen zu Bereichen von Parametern 2

Mit welcher Wahrscheinlichkeit liegt der Parameter zwischen 0.5 und 0.75?

```
samples %>%
  filter(p_grid > .5 & p_grid < .75) %>%
  summarise(sum     =       n() / 1e4,
            percent = 100 * n() / 1e4)  # In Prozent
```

```
## # A tibble: 1 × 2
##     sum percent
##   <dbl>   <dbl>
## 1 0.558    55.8
```

Mit welcher Wahrscheinlichkeit liegt der Parameter zwischen 0.9 und 1?

```
samples %>%
  filter(p_grid >= .9 & p_grid <= 1) %>%
  summarise(sum     =       n() / 1e4,
            percent = 100 * n() / 1e4)  # In Prozent
```

```
## # A tibble: 1 × 2
##     sum percent
##   <dbl>   <dbl>
## 1     0       0
```

Fragen zu Bereichen von Wahrscheinlichkeitsmassen

- Schätzbereiche von Parameterwerten nennt man auch *Konfidenz- oder Vertrauensintervall* (synonym: *Kompatibilitätsintervall* oder *Passungsbereich*).

Welcher Parameterwert wird mit 80% Wahrscheinlichkeit nicht überschritten, laut unserem Modell? (Gesucht sind also die unteren 80% Posteriori-Wahrscheinlichkeit)

```
samples %>%
  summarise(quantil80 =
              quantile(p_grid, p = .8))
```

```
## # A tibble: 1 × 1
##   quantil80
##       <dbl>
## 1     0.778
```

Was ist das *mittlere* Intervall, das mit 80% Wahrscheinlichkeit den Parameterwert enthält, laut dem Modell?

```
samples %>%
  summarise(
    quant_10 = quantile(p_grid, 0.1),
    quant_90 = quantile(p_grid, 0.9))
```

```
## # A tibble: 1 × 2
##   quant_10 quant_90
##      <dbl>    <dbl>
## 1    0.444    0.778
```

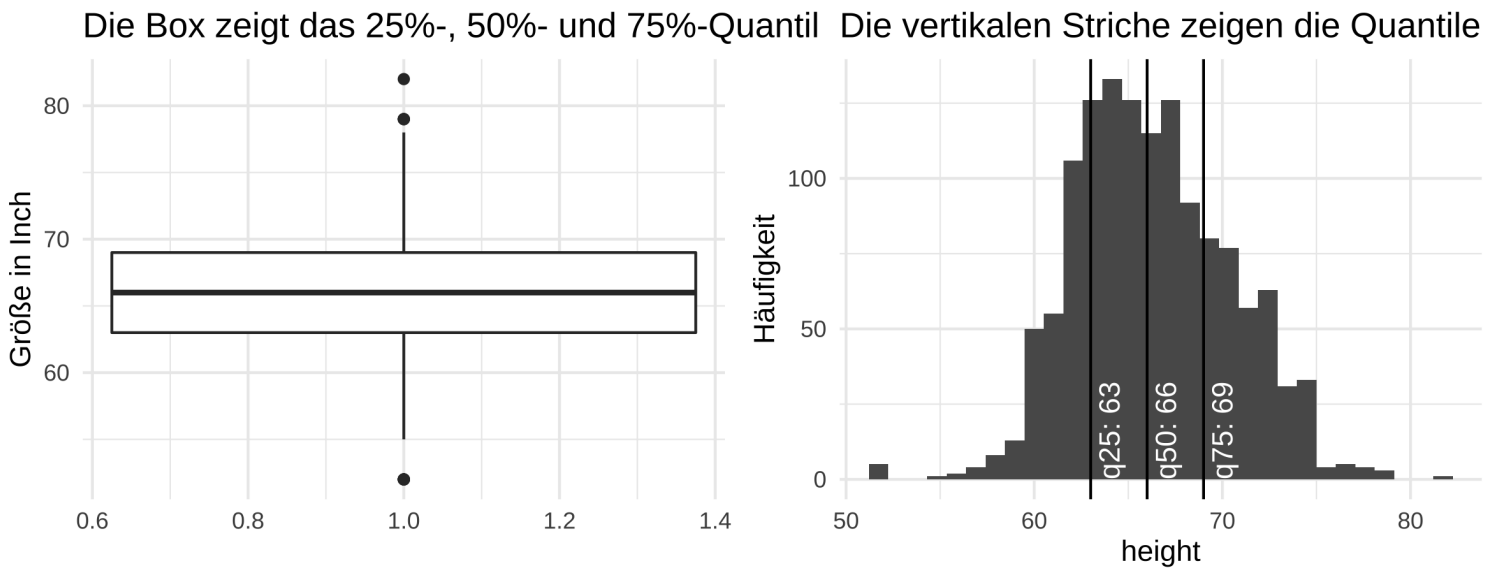
Solche Fragen lassen sich mit Hilfe von *Quantilen* beantworten.

Zur Erinnerung: Quantile

Beispiel: Wie groß sind die Studentis (Quelle des Datensatzes)? Das Quantil von z.B. 25% zeigt die Körpergröße der 25% kleinsten Studentis an, analog für 50%, 75%:

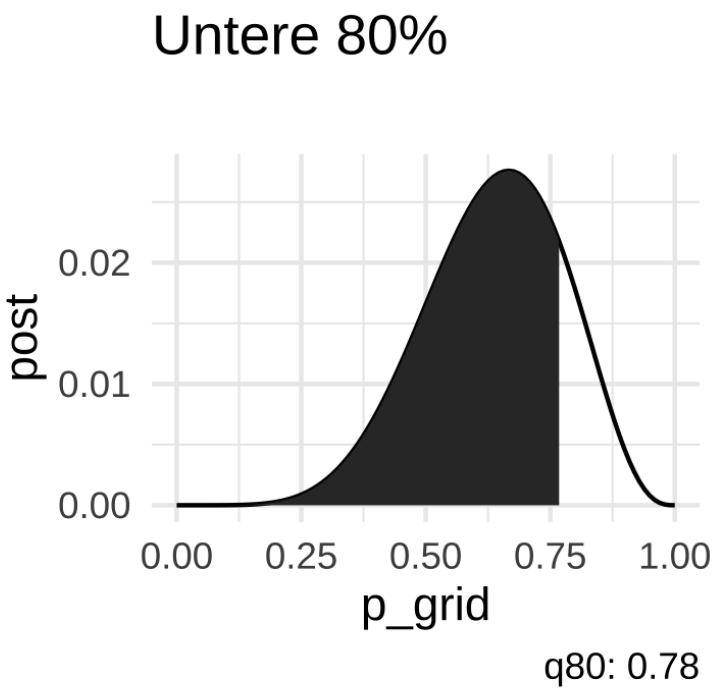
q25	q50	q75
63	66	69

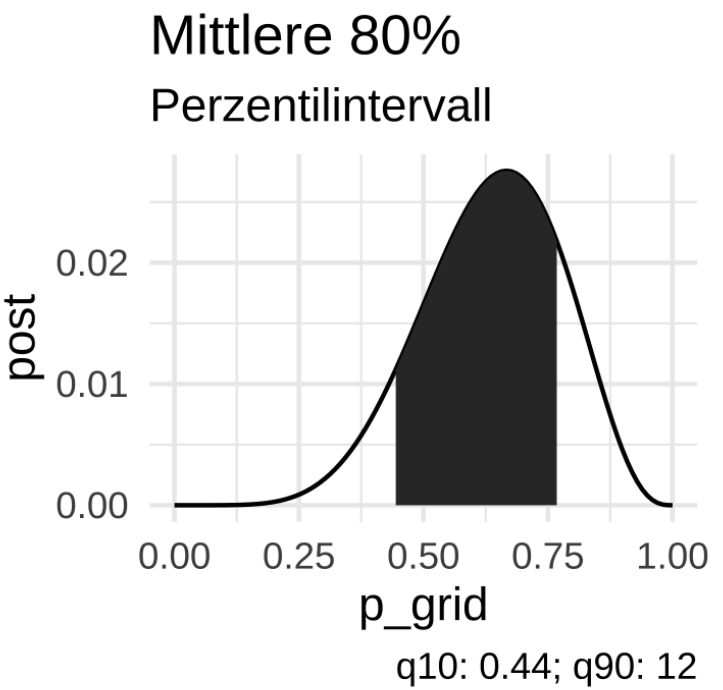
Visualisierung der Quantile:



Visualisierung der Massen-Intervalle

Intervalle (Bereiche), die die Wahrscheinlichkeitsmasse hälftig auf die beiden Ränder aufteilen, nennen wir *Perzentilintervalle*.





Schiefe Posteriori-Verteilungen sind möglich

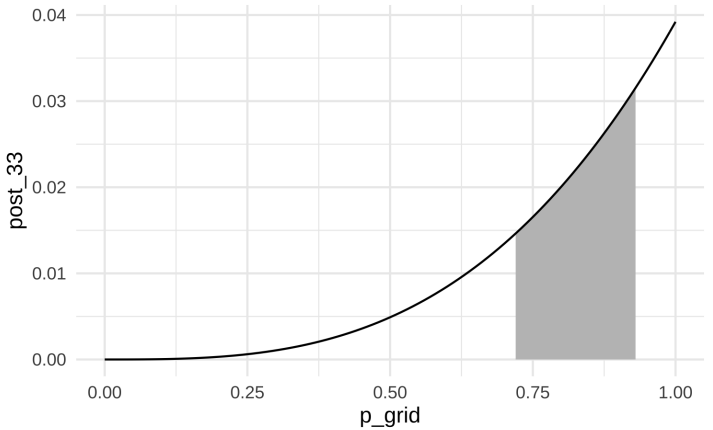
```
d_33 <-
  tibble(p_grid = seq(0,1, by =.01),
         prior = 1) %>%
  mutate(likelihood = dbinom(3, size = 3, prob = p_grid)) %>%
  mutate(unstand_post = likelihood * prior) %>%
  mutate(post_33  = unstand_post / sum(unstand_post))

samples_33 <-
  d_33 %>%
    slice_sample(n = 1e4,
                 weight_by = post_33,
                 replace = T)
```

p_grid	prior	likelihood	unstand_post
0.99	1	0.97	0.97
0.86	1	0.64	0.64
1.00	1	1.00	1.00
0.67	1	0.30	0.30
0.99	1	0.97	0.97
1.00	1	1.00	1.00

Intervalle höchster Dichte

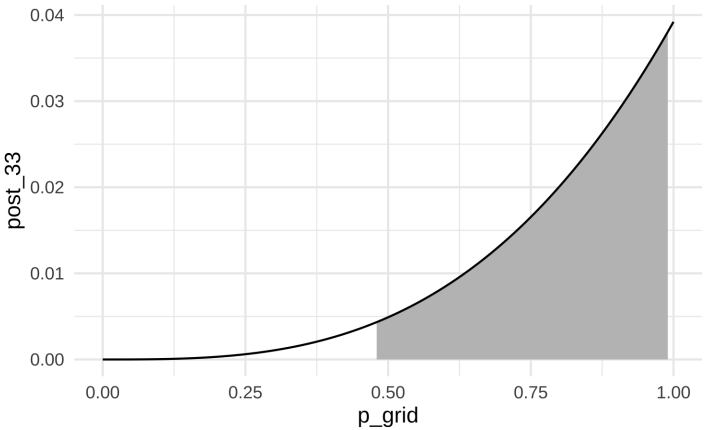
Daten: 3 Würfe mit 3 Treffern

50%-Perzentil-Intervall



Der wahrscheinlichste Parameterwert (1) ist *nicht* im Intervall enthalten!

50%-Intervall höchster Dichte



Der wahrscheinlichste Paramterwert (1) *ist* im Intervall enthalten!

Intervalle höchster Dichte vs. Perzentilintervalle

- Bei symmetrischer Posteriori-Verteilung sind beide Intervalle ähnlich
- Perzentilintervalle sind verbreiteter
- *Intervalle höchster Dichte* (Highest Posteriori Density Interval, HPDI) sind bei schiefen Post-Verteilungen zu bevorzugen
- Intervalle höchster Dichte sind die *schmalsten* Intervalle für eine gegebene Wahrscheinlichkeitsmasse

Intervallbreite HDPI: 0.16

```
rethinking::HPDI(samples$p_grid, prob = .5)
```

```
##      |0.5      0.5|
## 0.5555556 0.6666667
```

Intervallbreite PI: 0.23 (Quantile)

```
rethinking::PI(samples$p_grid, prob = .5)
```

```
##       25%       75%
## 0.5555556 0.7777778
```

Punktschätzungen

Datendatz `samples`, 6 Treffer bei 9 Würfen.

Lageparameter

Z.B. Welchen mittleren Wasseranteil muss man annehmen?

```
library(tidybayes)
samples %>%
  summarise(
    mean   = mean(p_grid),
    median = median(p_grid),
    # Mode() ist aus tidybayes:
    modus = Mode(p_grid))
```

```
## # A tibble: 1 × 3
##    mean median modus
##   <dbl>  <dbl> <dbl>
## 1 0.636  0.667 0.667
```

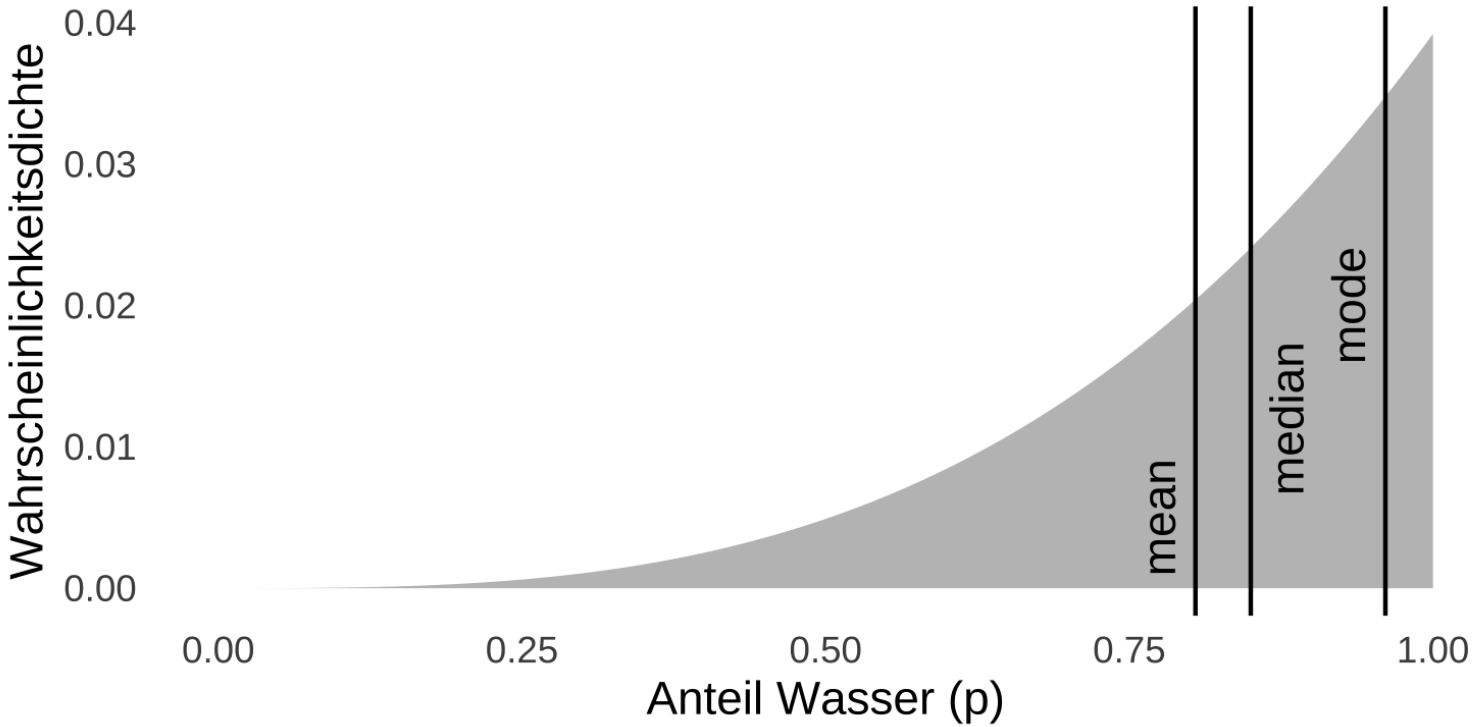
Streuungsparameter

Z.B. "Wie unsicher sind wir in der Schätzung des mittleren Wasseranteils?"

```
samples %>%
  summarise(
    p_sd   = sd(p_grid),
    p_iqr = IQR(p_grid),
    p_mad = mad(p_grid))
```

```
## # A tibble: 1 × 3
##    p_sd p_iqr p_mad
##   <dbl> <dbl> <dbl>
## 1 0.138 0.222 0.165
```

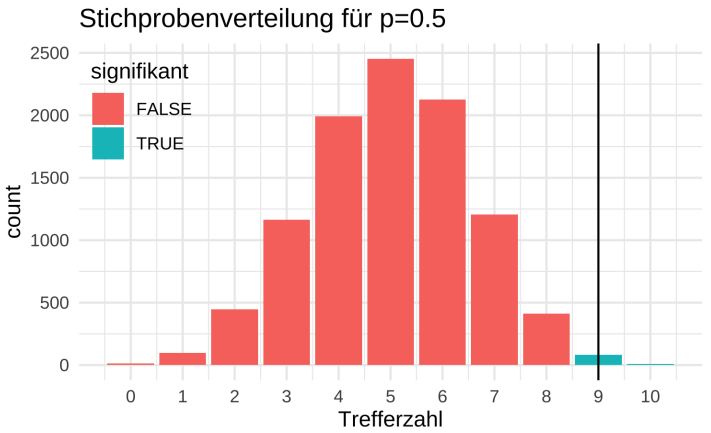
Visualisierungen der Punktschätzer

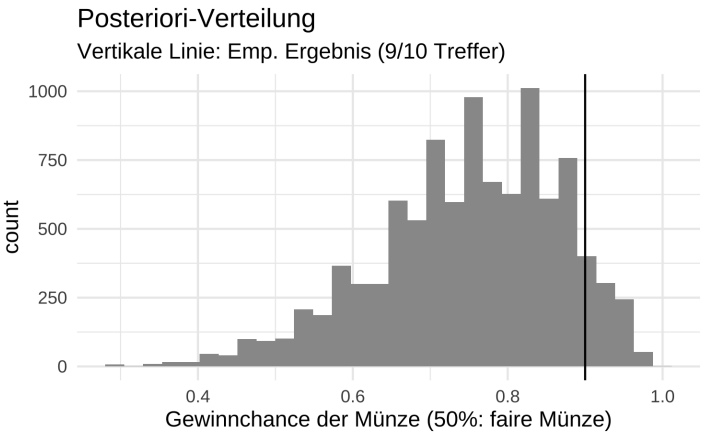


Je symmetrischer die Verteilung, desto näher liegen die Punktschätzer aneinander (und umgekehrt).

Der zwielichte Dozent: Stichproben-Vert. vs. Post-Vert.

Daten: 9 von 10 Treffern beim Münzwurf. Ist die Münze fair?





Die *Stichprobenverteilung* zeigt, wie Wahrscheinlich der empirischen Daten D (z.B. 9 von 10 Treffer) ist *gegeben* eines Parameterwerts p (z.B. $p = 0.5$): $Pr(D|p)$.

Die *Posteriori-Verteilung* gibt die Wahrscheinlichkeit jedes Parameterwerts p wider, gegeben der empirischen Daten D: $Pr(p|D)$.

Die meisten Forschungsfragen lassen sich mit der Post-Verteilung beantworten, nicht mit der Stichprobenverteilung.

Mit Stichproben neue Beobachtungen simulieren

Wir simulieren die Wasserzahl bei Globuswürfen

Likelihood (L): Wahrscheinlichkeit für $w = 0, 1, 2$ bei $N = 2$ und $p = 0.7$:

```
L <- dbinom(0:2, size = 2, prob = 0.7)
```

```
## [1] 0.09 0.42 0.49
```

Wir simulieren $n = 1$ neuen Globusversuch mit $N = 2, p = 0.7$ und zählen die (Wasser-)Treffer:

```
rbinom(n = 1, size = 2, prob = .7)  # 0 Treffer (Wasser)
```

```
## [1] 0
```

Warum nicht $n = 10$ neue Globusversuche simulieren:

```
rbinom(n = 10, size = 2, prob = 0.7)
```

```
##  [1] 0 2 1 1 1 1 2 1 1 2
```

Diese Versuche geben Aufschluss, welche Daten (wie oft Wasser) man bei einem bestimmten Modell, p, N, erwarten kann.

Never trust a Golem

from Imgflip Meme Generator

Quelle: https://imgflip.com/i/5qmhmo

Traue niemals einem Golem (einem Modell)

Immer prüfen und wachsam bleiben:

- (Inwieweit) decken sich die simulierten Daten mit den tatsächlichen Beobachtungen?
- Wie realistisch sind die Modellannahmen?
- Kann man das Modell aus verschiedenen Perspektiven prüfen?

Mit guten Simulationen kommt man den wahren Werten nahe

Warum nicht $n = 10^6$ neue Globusversuche simulieren:

```
d <-
  tibble(
    draws =
      rbinom(1e6,
             size = 2,
             prob = .7))
d %>%
  count(draws) %>%
  mutate(proportion =
           n / nrow(d))
```

```
## # A tibble: 3 × 3
##   draws      n proportion
##   <int>  <int>      <dbl>
## 1     0  89770     0.0898
## 2     1 420629     0.421
## 3     2 489601     0.490
```

Diese simulierten Häufigkeiten sind sehr ähnlich zu den theoretisch bestimmten Häufigkeiten mit `dbinom`: Unser Modell liefert plausible Vorhersagen.

```
dbinom(0:2, size = 2, prob = .7)
```

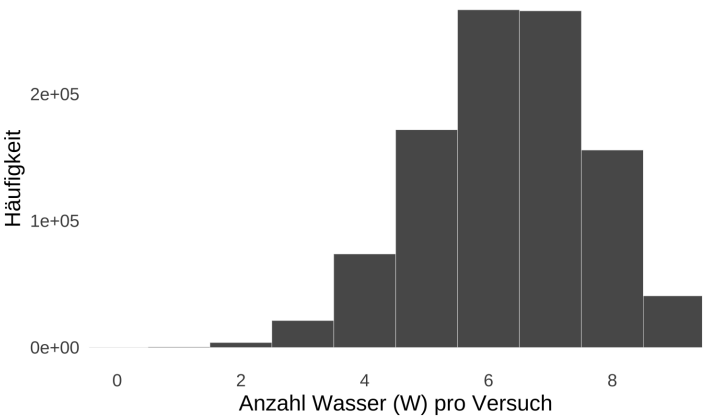
```
## [1] 0.09 0.42 0.49
```

Stichprobenverteilung

Wir ziehen viele ($n=10^6$) Stichproben für den Versuch $N = 9$ Globuswürfe mit $p = 0.7$.

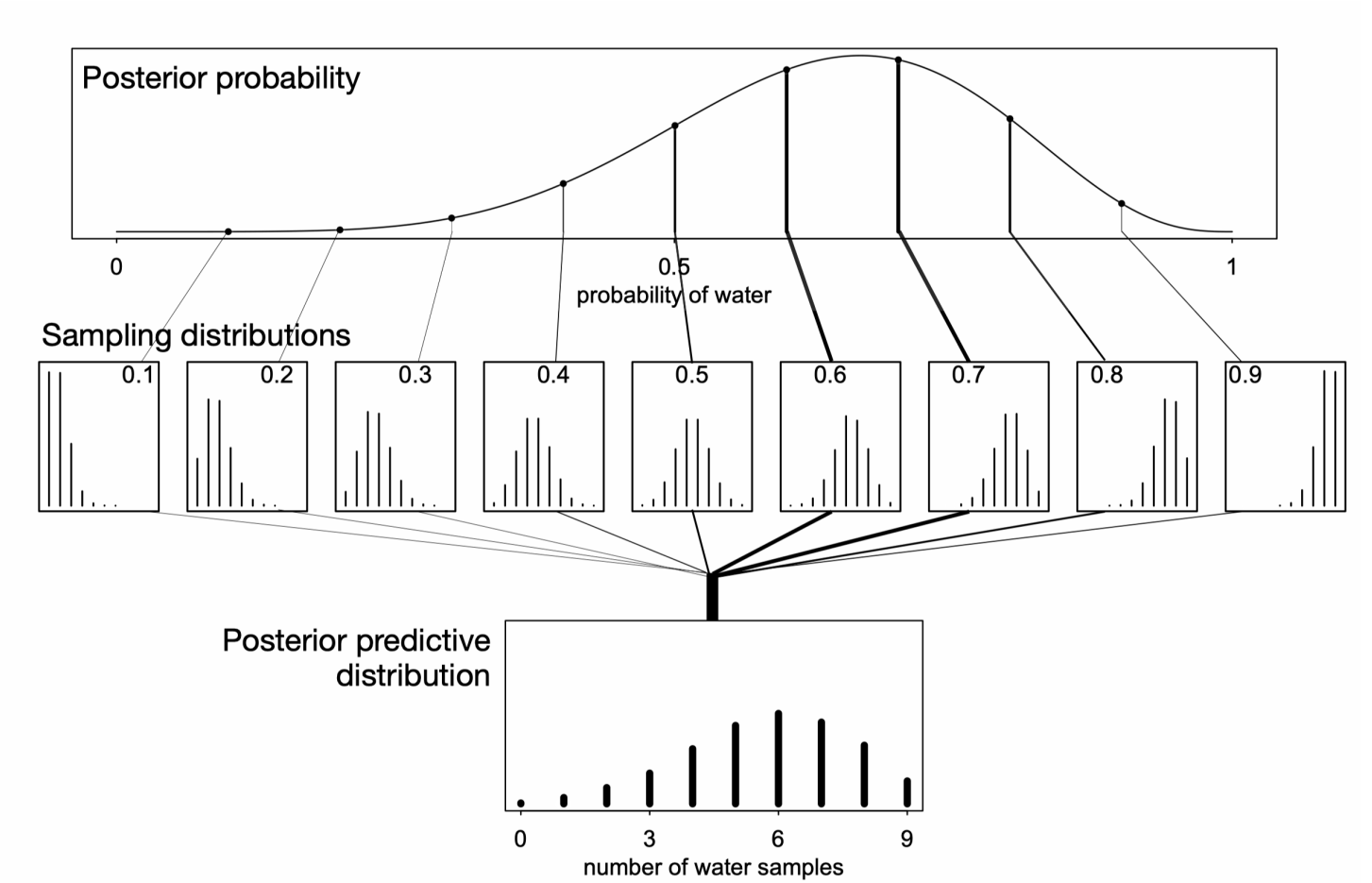
Wie viele Wasser (W) erhalten wir wohl typischerweise?

```
n_draws <- 1e6

d <-
  tibble(draws =
           rbinom(
             n_draws,
             size = 9,
             prob = .7
           ))

plot1 <-
d %>%
  ggplot(aes(x = draws)) +
  geom_histogram()
```



Die *Stichprobenverteilung* zeigt, welche Stichprobendaten laut unserem Modell zu erwarten sind. Wir können jetzt prüfen, ob die echten Daten zu den Vorhersagen des Modells passen.

Visualisierung der PPV



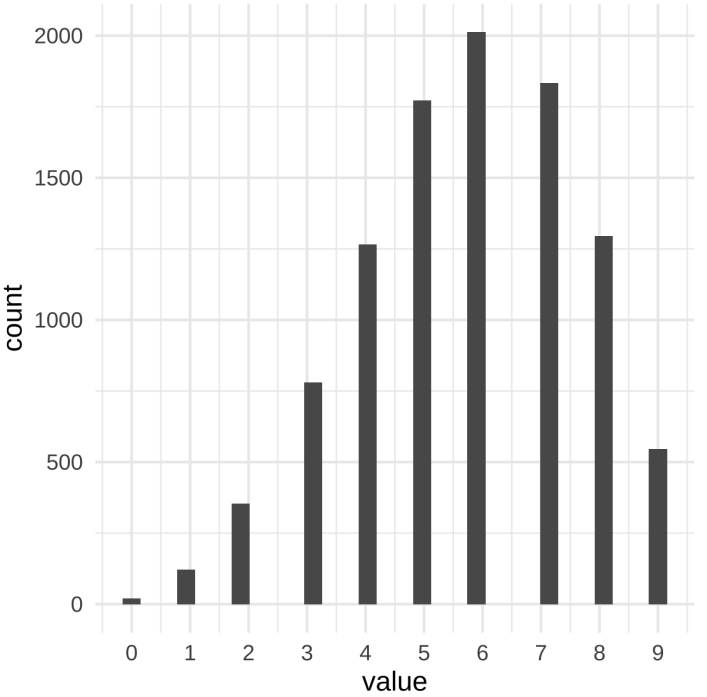
Quelle: McElreath (2020)

So viele Verteilungen...

- Die *Posteriori-Verteilung* gibt Aufschluss zur Häufigkeit (Wahrscheinlichkeit) von Parameterwerten:
  - Wie wahrscheinlich ist es, dass "in Wirklichkeit" der Wasseranteil 70% beträgt, also $\pi = .7$
  - In der Wissenschaft ist man meist an den Parametern interessiert.
- Die *PPV* gibt Aufschluss zur Häufigkeit von neuen Beobachtungen:
  - Welche Beobachtungen (wie viele Wasser/Treffer) sind in Zukunft, bei erneuter Durchführung, zu erwarten.
  - Für die Praxis kann das eine interessante Frage sein.
- Der *Likelihood* gibt Aufschluss, wie gut eine bestimmte Hypothese die Datenlage erklärt.
  - Wie gut passt die Hypothese $\pi = 0.7$ auf die Datenlage 6 von 9 Treffern beim Globusversuch?
  - Der Likelihood kann aus der Stichprobenverteilung herausgelesen werden.

PPV berechnen

```
ppv <-
  rbinom(1e4,
         size = 9,
         prob = samples$p_grid) %>%
  as_tibble()

ppv_plot2 <-
  ppv %>%
  ggplot() +
  aes(x = value) +
  geom_histogram() +
  scale_x_continuous(
    breaks = 0:9)
```



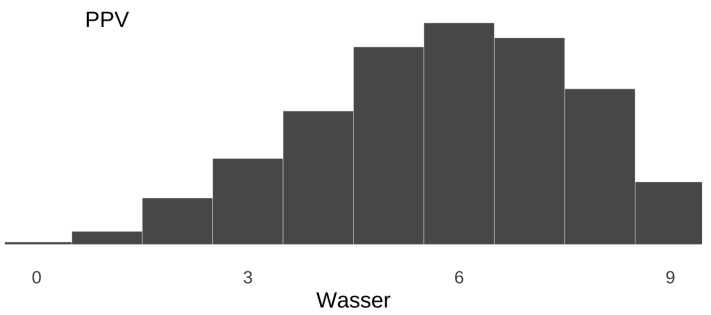
- Die PPV unseres Modells zeigt uns, dass wir in künftigen Versuchen zumeist 6 Treffer zu erwarten haben.
- Aber ein relativer breiter Bereich an Treffern ist ebenfalls gut laut unserer PPV erwartbar.

Vorhersagen sind schwierig

... gerade wenn sie die Zukunft betreffen, so ein Sprichtwort.

Das zeigt uns die PPV: Der PPV unseres Modells gelingt es zwar, der theoretisch wahrscheinlichste Parameterwert ist auch der häufigste in unseren Stichproben, aber die Vorhersagen haben eine große Streuung, birgt also hohe Ungewissheit.

Die PPV zeigt also, welche Beobachtungen laut unserem Modell künftig zu erwarten sind.



Würde man die Vorhersagen nur anhand eines bestimmten Parameterwertes (z.B $p = 0.6$) vornehmen, hätten die Vorhersagen zu wenig Streuung, würden also die Ungewissheit nicht ausreichend abbilden (Übergewissheit, Overconfidence).

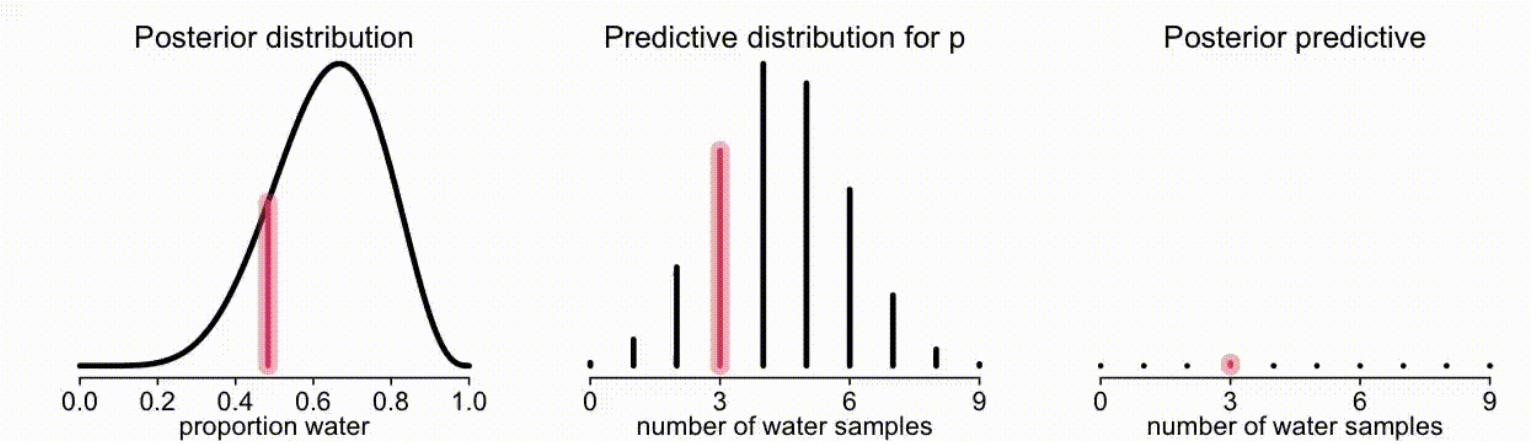
Zwei Arten von Ungewissheit in Vorhersagen von Modellen

1. *Ungewissheit innerhalb des Modells*: Auch wenn der (oder die) Modellparameter eines Modells mit Sicherheit bekannt sind, so bleibt Unsicherheit, welche Beobachtung eintreten wird: Auch wenn man sicher weiß, dass $p = 1/4$ Murmeln blau sind, so kann man nicht sicher sagen, welche Farbe die nächste Murmel haben wird (Ausnahme: $p = 1$ oder $p = 0$).
2. *Ungewissheit in den Modellparametern*: Wir sind uns nicht sicher, welchen Wert p (bzw. die Modellparameter) haben. Diese Unsicherheit ist in der Post-Verteilung dargestellt.

Um zu realistischen Vorhersagen zu kommen, möchte man beide Arten von Ungewissheit berücksichtigen: Das macht die *Posteriori-Prädiktiv-Verteilung (PPV)*.

Die PPV zeigt, welche Daten das Modell vorhersagt (prädiktiv) und mit welcher Häufigkeit, basierend auf der Post-Verteilung.

Vergleich der Verteilungen



- Links - *Posterior-Verteilung*: Wahrscheinlichkeiten der Parameterwerte
- Mitte - *Stichprobenverteilung*: Wahrscheinlichkeiten der Beobachtungen gegeben eines bestimmten Parameterwertes
- Rechts - *Posterior-Prädiktiv-Verteilung*: Wahrscheinlichkeiten der Beobachtungen unter Berücksichtigung der Unsicherheit der Posteriori-Verteilung

Quelle: R. McElreath

Hinweise

Zu diesem Skript

- Dieses Skript bezieht sich auf folgende Lehrbücher:
  - *Statistical Rethinking* (2. Auflage), Kapitel 3, McElreath (2020)
  - Der R-Code stammt aus Kurz (2021).
- Dieses Skript wurde erstellt am 2021-10-25 12:24:38 (WiSe 21).
- Lizenz: CC-BY
- Autor ist Sebastian Sauer.
- Um diese HTMLM-Folien korrekt darzustellen, ist eine Internet-Verbindung nötig.
- Eine PDF-Version kann erzeugt werden, indem man im Chrome-Browser druckt (Drucken als PDF).
- Mit der Taste ? bekommt man eine Hilfe über Shortcuts.

Literatur

Kurz, A. S. (2021). *Statistical rethinking with brms, ggplot2, and the tidyverse: Second edition*. URL: https://bookdown.org/content/4857/ (visited on Sep. 08, 2021).

McElreath, R. (2020). *Statistical rethinking: a Bayesian course with examples in R and Stan*. 2nd ed. CRC texts in statistical science. Boca Raton: Taylor and Francis, CRC Press. ISBN: 978-0-367-13991-9.