Gaussmodelle: Schätzen einer metrischen Variablen

Kapitel 4

Gliederung

1. Teil 1: Verteilungen
2. Teil 2: Gauss-Modelle: Wie groß sind die !Kung San?
3. Hinweise

Software

Für dieses Thema benötigen Sie einige R-Pakete, die Sie wie folgt installieren können:

```
pakete <- c("tidyverse", "rstan", "rstanarm", "bayesplot")

install.packages(pakete)
```

Für `rstan` wird weitere Software benötigt.

Verteilungen

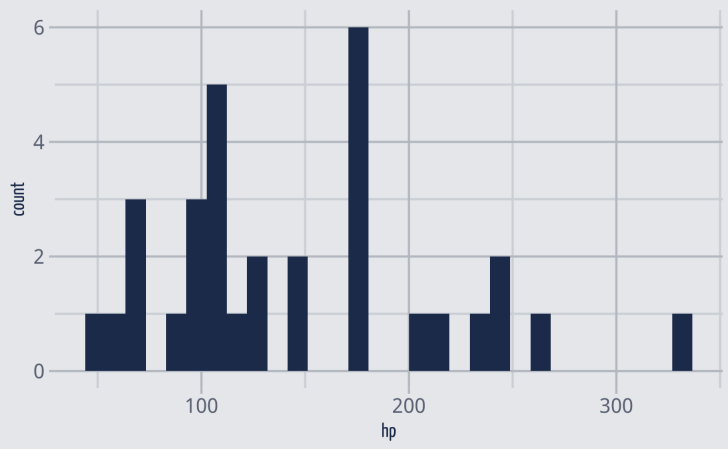
Häufigkeitsverteilung

Die Verteilung eines *diskreten* Merkmals X mit k Ausprägungen zeigt, wie häufig die einzelnen Ausprägungen sind.

```
  data(mtcars)
  mtcars %>%
    count(cyl)
```

```
##   cyl  n
## 1   4 11
## 2   6  7
## 3   8 14
```

Ein *stetiges* Merkmal lässt sich durch Klassenbildung diskretisieren:

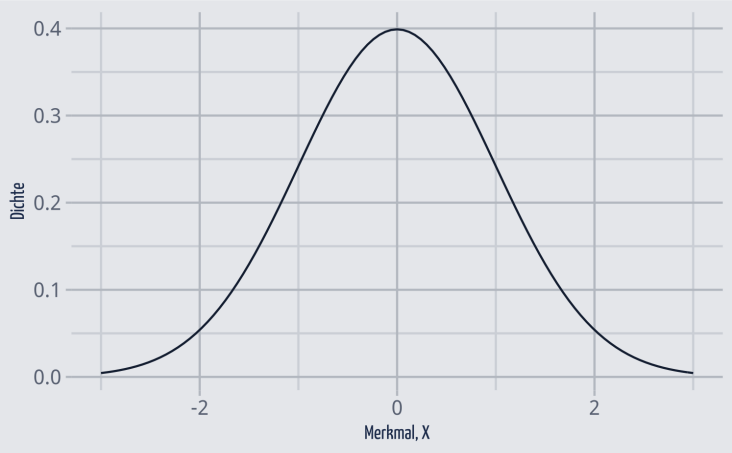


Wahrscheinlichkeitsverteilung

Eine *diskrete* Wahrscheinlichkeitsverteilung des Merkmals X ordnet jeder der k Ausprägungen $X = x$ eine Wahrscheinlichkeit p zu. So hat die Variable *Geschlecht eines Babies* die beiden Ausprägungen *Mädchen* und *Junge* mit den Wahrscheinlichkeiten $p_M = 51.2\%$ bzw. $p_J = 48.8\%$ (Gelman, Hill, and Vehtari, 2021).

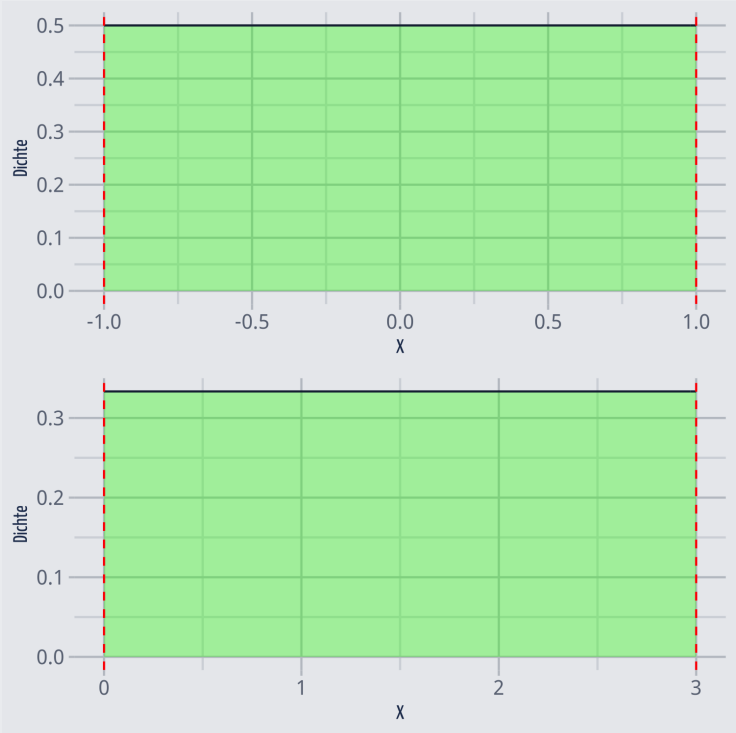
Bei *stetigen* Merkmalen X geht man von unendlich vielen Ausprägungen aus; die Wahrscheinlichkeit einer bestimmten Ausprägung ist (praktisch) Null: $p(X = x_j) = 0, \quad j = 1, \ldots, k$. Daher gibt man stattdessen die *Dichte* der Wahrscheinlichkeit an: Das ist die Wahrscheinlichkeit(smasse) pro eine Einheit von X.

Beispiele für Wahrscheinlichkeitsdichte



Bei $X = 0$ hat eine Einheit von X die Wahrscheinlichkeitsmasse von 40%.

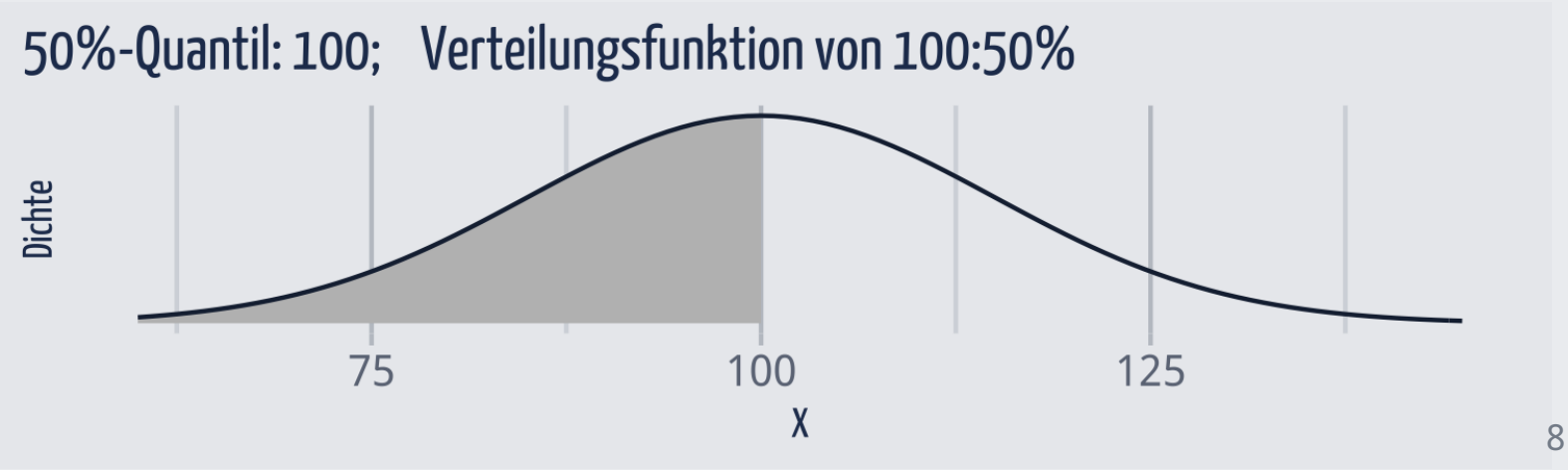
In Summe liegen 100% der Wahrscheinlichkeitsmasse unter der Kurve.



Bei $X = 0$ hat eine Einheit von X die Wahrscheinlichkeitsmasse von 50% bzw. 33%.

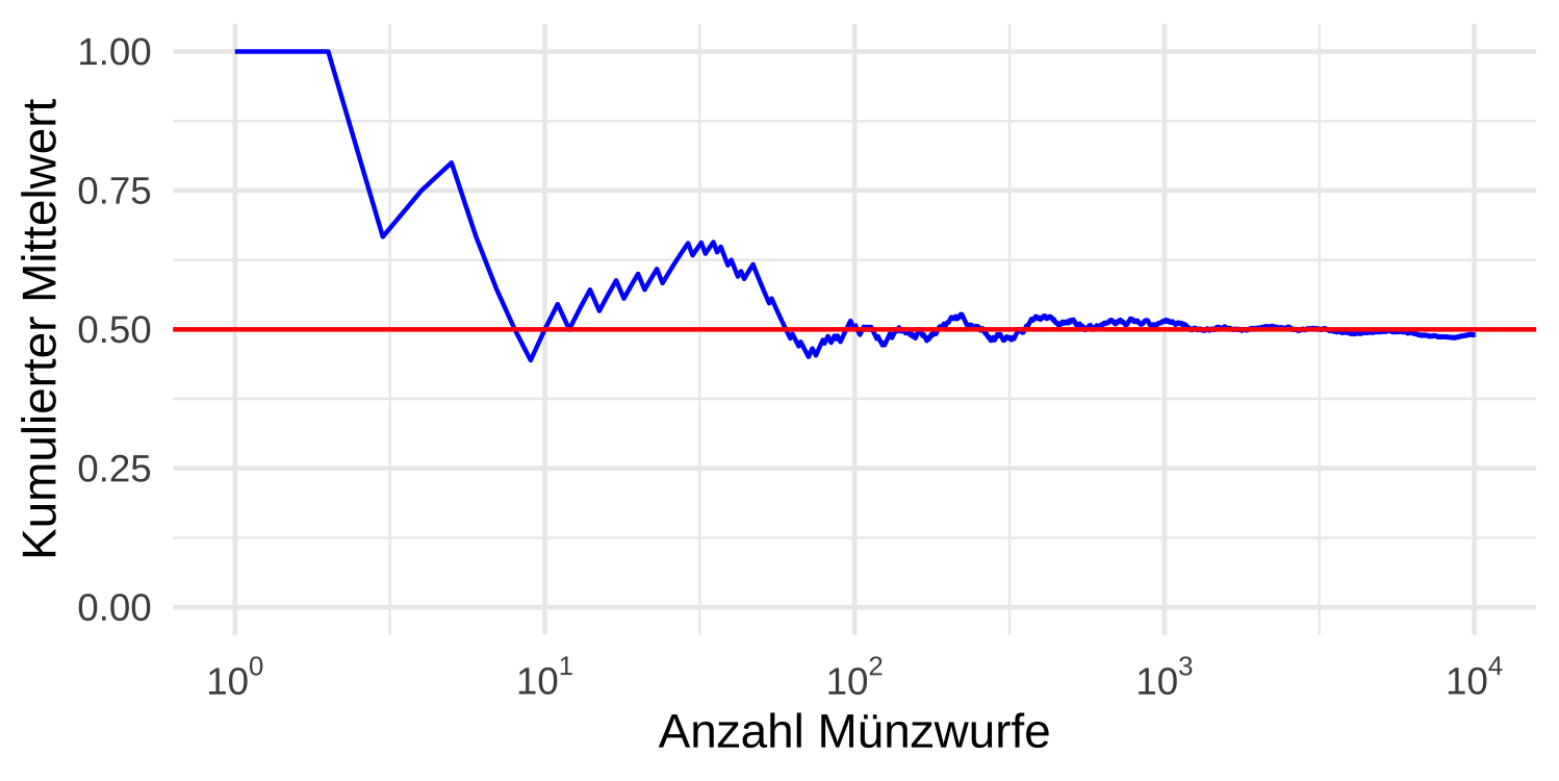
Quantile und Verteilungsfunktion

- *Quantile* teilen eine Verteilung so ein, dass ein Anteil p kleiner und der andere Teil $1 - p$ größer oder gleich dem Quantil q ist.
    - *Beispiel*: "50%-Quantil = 100" meint, dass 50% der Werte der Verteilung einen Wert kleiner als 100 haben.
- Die *Verteilungsfunktion F* für ein Quantil q gibt den Anteil der Verteilung an, der nur Werte höchstens so groß wie q beinhaltet. Sie zeigt also die kumulierte Wahrscheinlichkeit $[-\infty, q)$.
    - *Beispiel*: "F(100) = 50%" meint, dass der Anteil der Verteilung für Werte nicht größer als 100 50% beträgt.



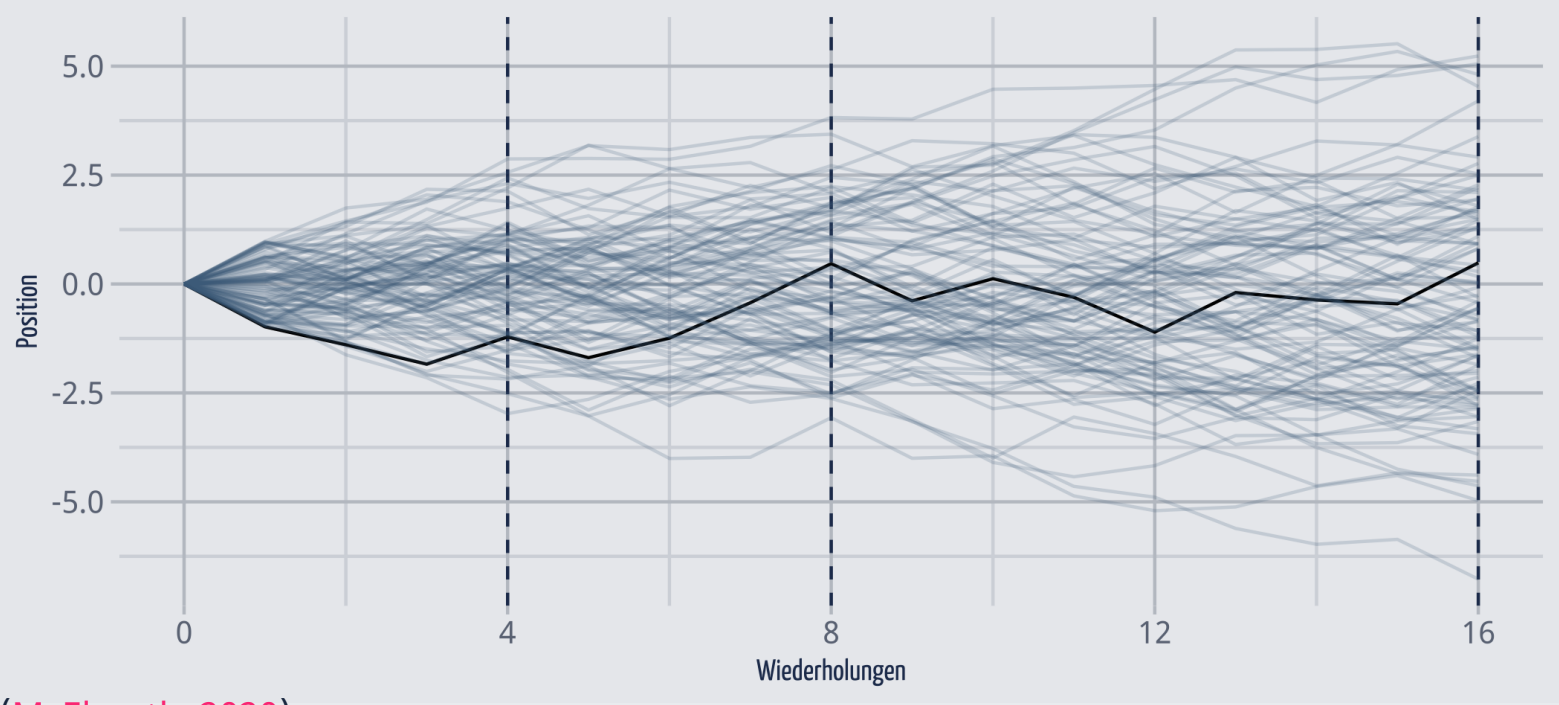
Gesetz der großen Zahl

Zieht man (zufällig) immer mehr Werte aus einer Verteilung (mit endlichem Mittelwert), nähert sich der Mittelwert der Stichprobe immer mehr mit dem Mittelwert (oft als Erwartungswert bezeichnet) der Verteilung an (Taleb, 2019)



Normal auf dem Fußballfeld

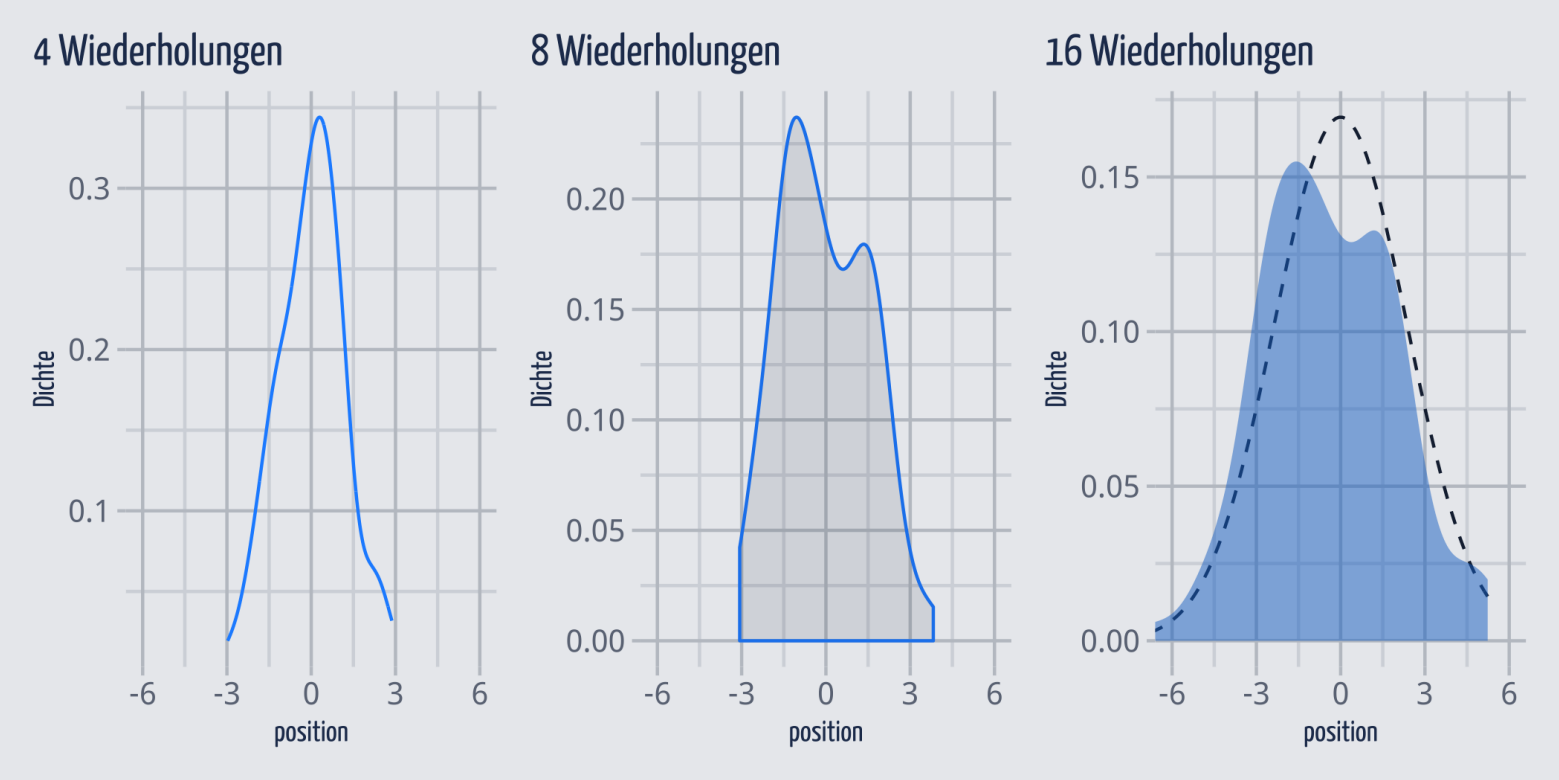
Sie und 1000 Ihrer besten Freunde stehen auf der Mittellinie eines Fußballfelds (eng). Auf Kommando werfen alle jeweils eine Münze; bei Kopf geht man einen Schritt nach links, bei Zahl nach rechts. Das wird 16 Mal wiederholt. Wie wird die Verteilung der Positionen wohl aussehen?



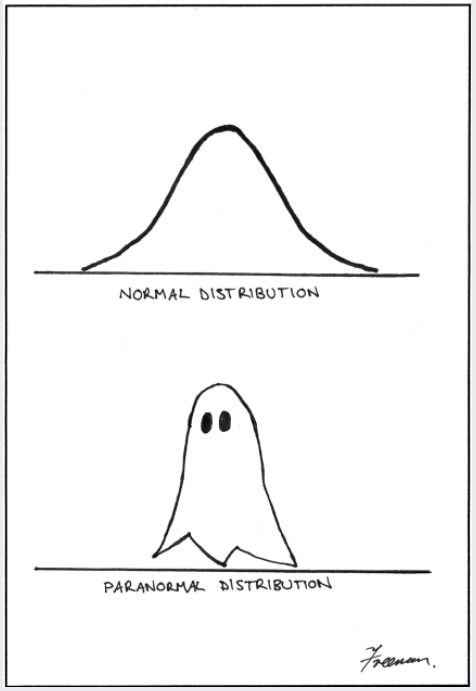
(McElreath, 2020)

Normal durch Addieren

Die Summe vieler (gleich starker) Zufallswerte (aus der gleichen Verteilung) erzeugt eine Normalverteilung; egal aus welcher Verteilung die Zufallswerte kommen (Zentraler Grenzwertsatz).

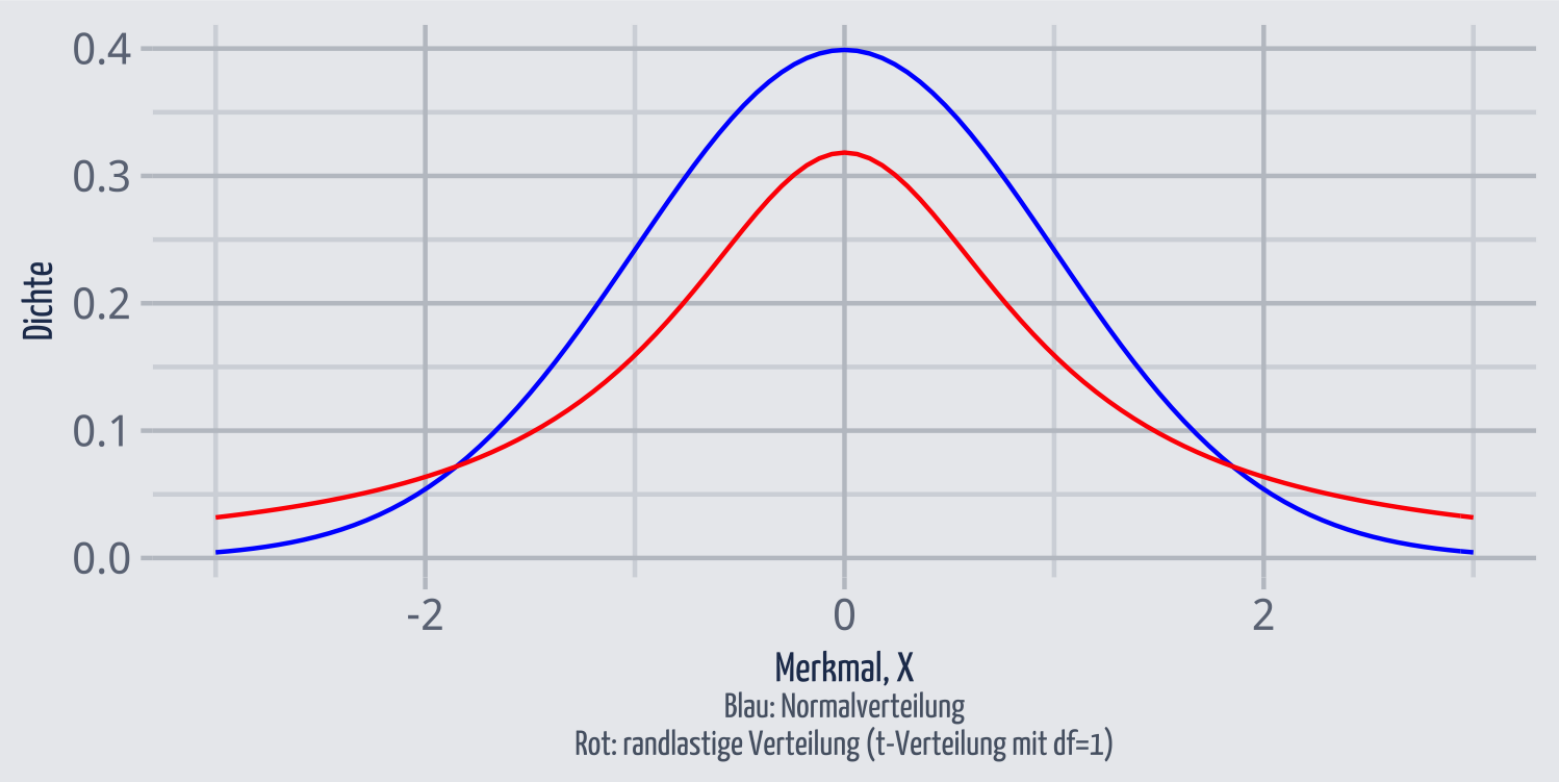


Nicht verwechseln



(Freeman, 2006)

Normalverteilung vs. randlastige Verteilungen



Bei randlastigen Verteilungen ("fat tails") kommen Extremereignisse viel häufiger vor als bei Normalverteilungen. Deshalb ist es wichtig sein, zu wissen, ob eine Normalverteilung oder eine randlastige Verteilung vorliegt. Viele statistische Methoden sind nicht zuverlässig bei (stark) randlastigen Methoden (Taleb, 2019)

Beispiele für Normal- und randlastige Verteilungen

Normal verteilt

- Größe
- Münzwürfe
- Gewicht
- IQ
- Blutdruck
- Ausschuss einer Maschine

Randlastig verteilt

- Vermögen
- Verkaufte Bücher
- Ruhm
- Aktienkurse
- Erdbeben
- Pandemien
- Kriege
- Erfolg auf Tinder
- Meteoritengröße
- Stadtgrößen

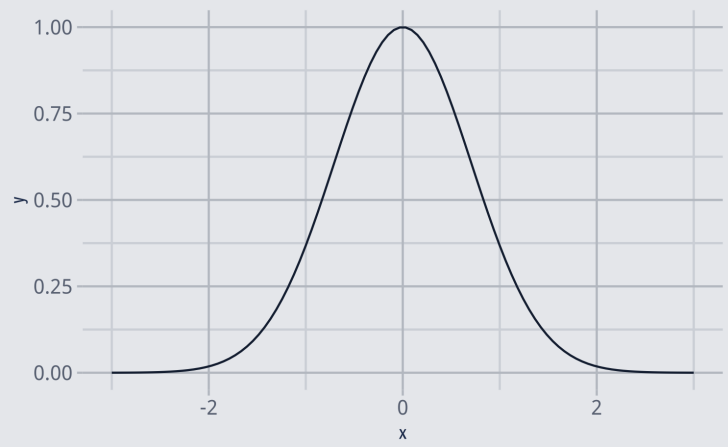
Formel der Normalverteilung

Vereinfacht ausgedrückt lässt die Normalverteilung $\mathcal{N}$ durch Exponenzieren einer Quadratfunktion beschreiben:

$$\mathcal{N} \propto e^{-x^2}$$

mit $e = 2.71...$, der Eulerschen Zahl.

```
d <-
  tibble(
    x = seq(-3, 3,
            length.out = 100),
    y = exp(-x^2)
  )

d %>%
  ggplot() +
  aes(x = x, y = y) +
  geom_line()
```



Die Normalverteilung wird auch *Gauss-Verteilung* oder *Glockenkurve* genannt.

IQ-Verteilung: Quantile

$IQ \sim \mathcal{N}(100, 15)$

- Wie schlau muss man sein, um zu den unteren 75%, 50%, 25%, 5%, 1% zu gehören?
- Anders gesagt: Welcher IQ-Wert wird von 75%, 50%, ... der Leute nicht überschritten?

Ziehen wir Stichproben aus $\mathcal{N}(100, 15)$:

```
d <-tibble(
  iq = rnorm(1e4,
             mean = 100,
             sd = 15))

probs <- c(0.75,.5,.25,.05,.01)

d_summary <- d %>%
  summarise(
    p = probs,
    q = quantile(iq, probs))
```

p	q
0.75	110
0.50	100
0.25	90
0.05	75
0.01	65

Das *Quantil* `(q)` zur kumulierten Wahrscheinlichkeit `(p=75)` ist 110, etc.

IQ-Verteilung: Anteile

$IQ \sim \mathcal{N}(100, 15)$

- Welcher Anteil p gehört zu den IQ-Werten 75, 100, 115, 130?
- Anders gesagt: Welcher Anteil der Wahrscheinlichkeitsmasse der Verteilung liegt unter IQ=75, IQ=100, etc.?

Ziehen wir Stichproben aus $\mathcal{N}(100, 15)$:

```
d <-
  tibble(
    iq = rnorm(1e4,
               mean = 100,
               sd = 15)) %>%
  mutate(iq = round(iq))

qs <- c(75,100,115,130)

d %>%
  count(p_100 = iq < 100) %>%
  mutate(prop = n / sum(n))
```

p_100	n	prop
FALSE	5090	0.51
TRUE	4910	0.49

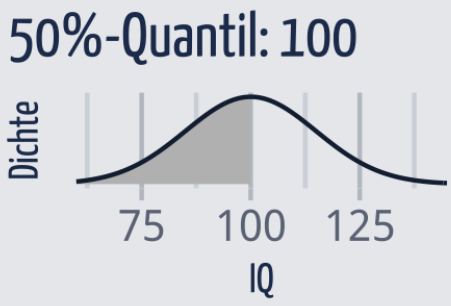
Anstelle von `iq < 100` kann man `iq < 115` einsetzen, etc.

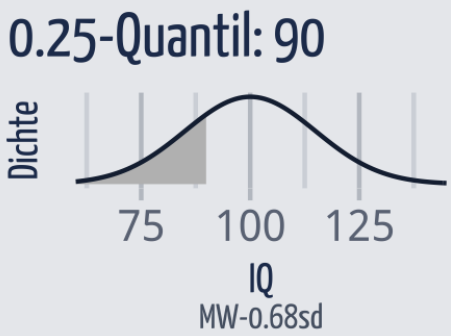
Die *Verteilungsfunktion* (der Anteil der Wahrscheinlichkeitsmasse), `p`, für IQ-Werte nicht größer als 100, d.h. zum Quantil `(q=100)`, ist 50%, etc.

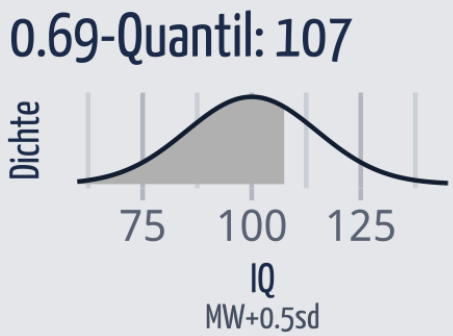
Quantile der Normalverteilung visualisiert

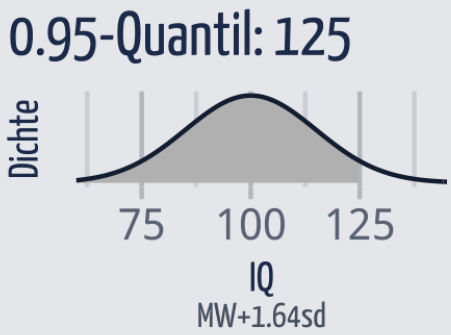
$IQ \sim \mathcal{N}(100, 15)$

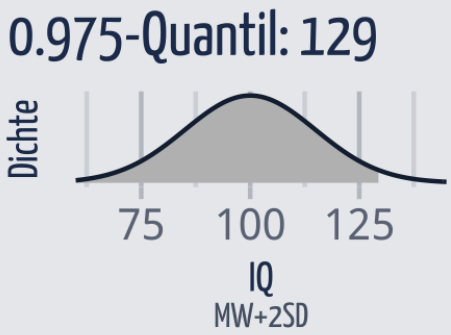
```r
qnorm(.50, mean = 100, sd = 15)  # 50%-Quantil
pnorm(100, mean = 100, sd = 15)  # Verteilungsfunktion für IQ=100
```

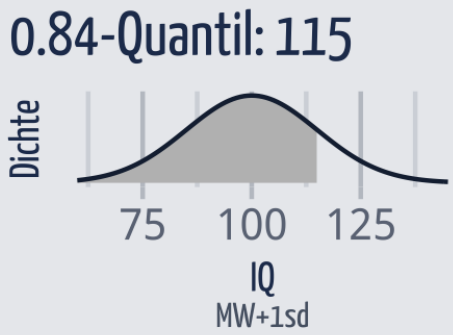












Normalverteilung als konservative Wahl

Uni Greifswald, Public domain,

via Wikimedia Commons

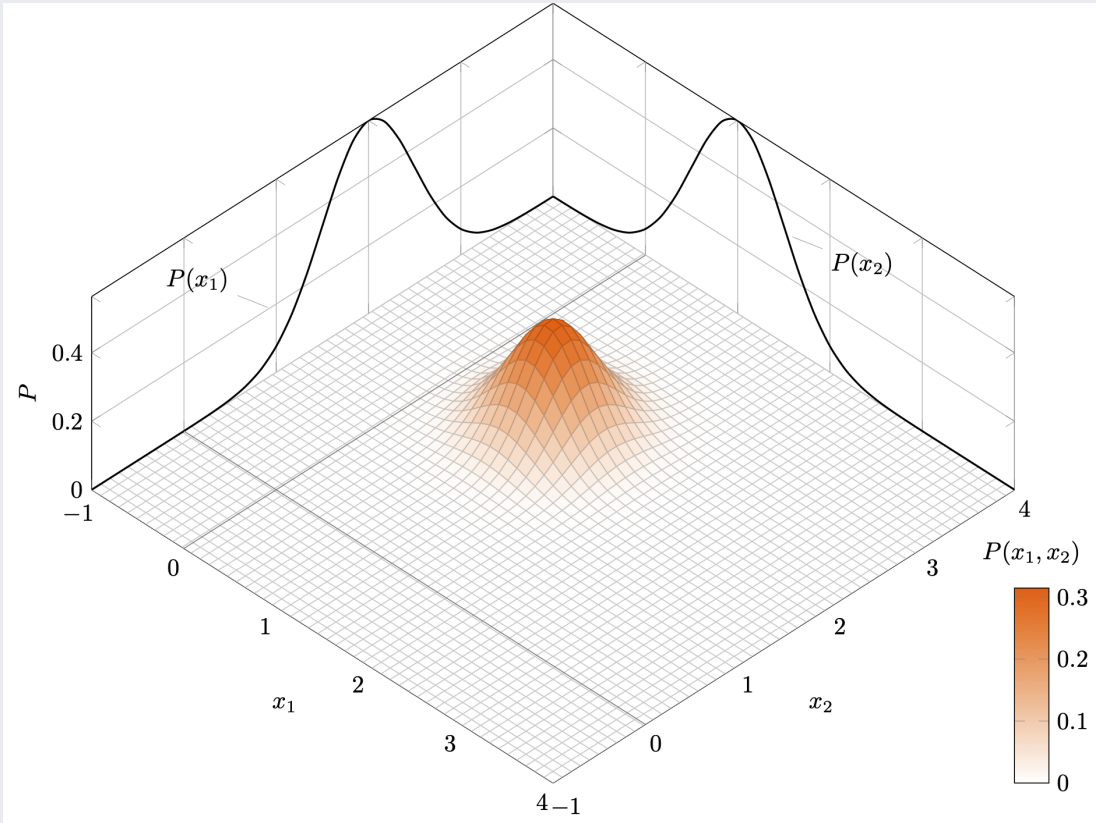
Ontologische Begründung

- Wirken viele, gleichstarke Einflüsse additiv zusammen, entsteht eine Normalverteilung (McElreath, 2020), Kap. 4.1.4.

Epistemologische Begründung

- Wenn wir nur wissen, dass eine Variable über einen endlichen Mittelwert und eine endliche Varianz verfügt und wir keine weiteren Annahmen treffen bzw. über kein weiteres Vorwissen verfügen, dann ist die Normalverteilung die plausibelste Verteilung (maximale Entropie) (McElreath, 2020), Kap. 7 und 10.

Zweidimensionale Normalverteilung, unkorreliert



Quelle

Vgl. auch dieses Diagramm

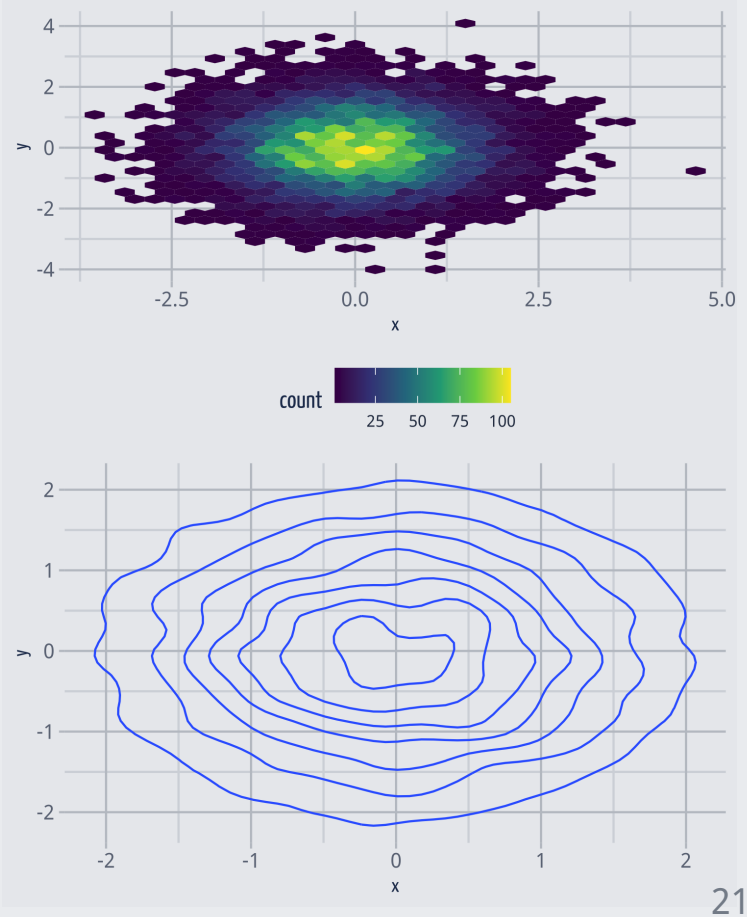
2D-Normalverteilung mit R, unkorreliert

$$r(X, Y) = 0$$

```
d1 <-
  tibble(
    x=rnorm(1e4),
    y=rnorm(1e4)
  )

ggplot(d1) +
  aes(x, y) +
  geom_hex()

ggplot(d1) +
  aes(x, y) +
  geom_density2d()
```

ggplot-Referenz, Quellcode

Mit `scale_fill_continuous(type = "viridis")` kann man die Farbpalette der Füllfarbe ändern.



2D-Normalverteilung mit R, korreliert, r=0.7

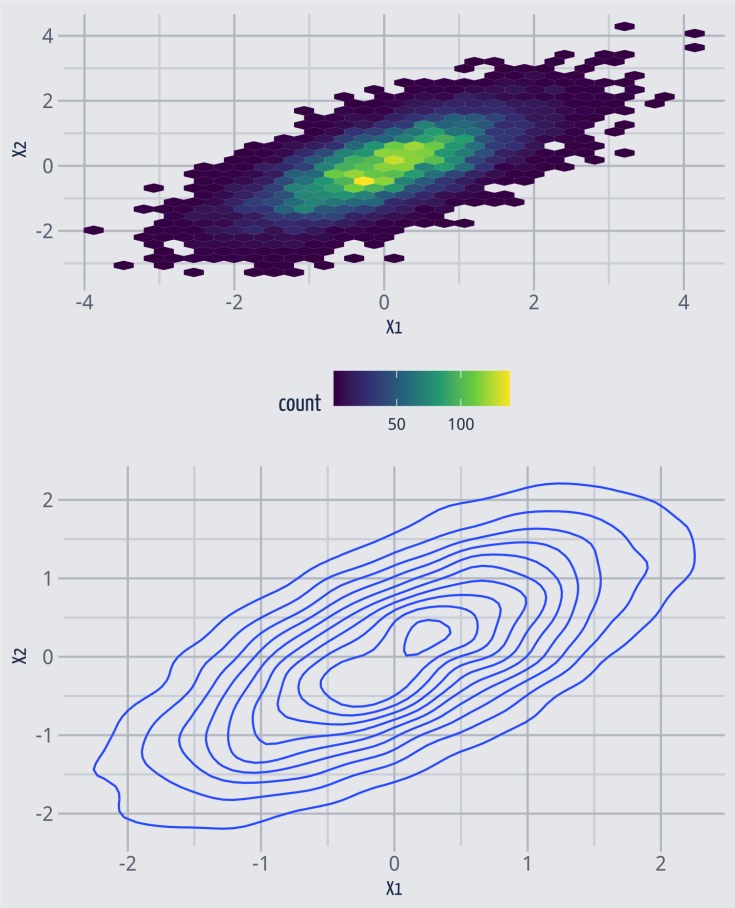
Die ersten paar Zeilen der Daten:

X1	X2
1.07	1.16
−0.15	−0.82
1.47	0.11

Berechnen wir die Korrelation `r`:

```
d2 %>%
  summarise(
    r = cor(X1,X2),
    n = n()
  )
```

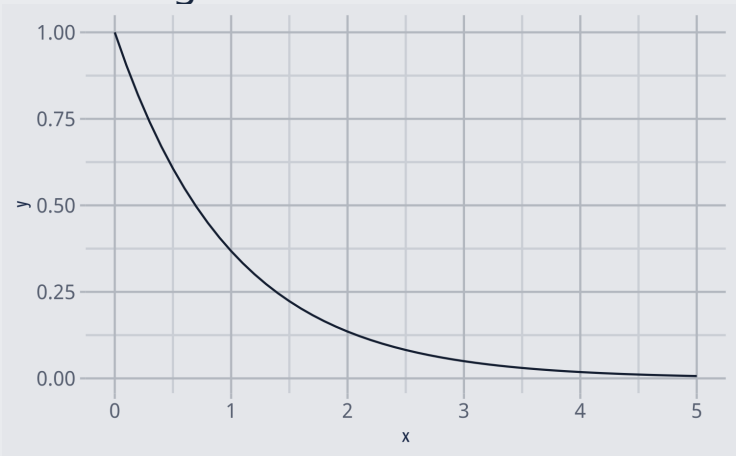
r	n
0.70	10,000.00



Die Mensch-ärgere-dich-nicht-Verteilung

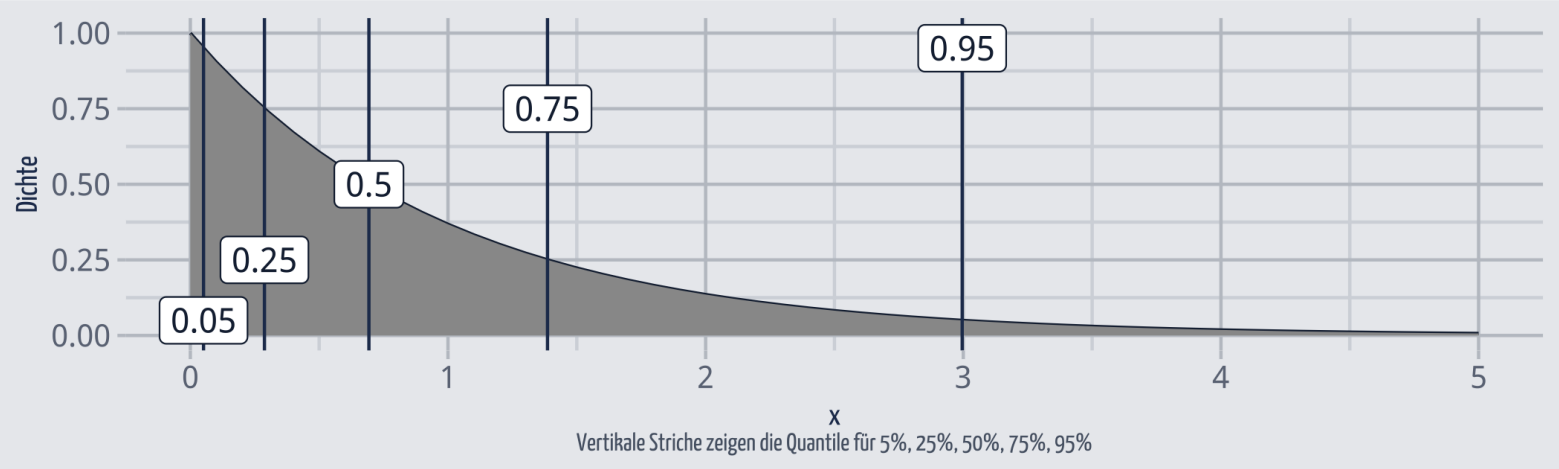
- Wie lange muss man warten, bis man bei Mensch-ärgere-dich-nicht raus darf?
- Wieviel Vitamine sind nach einer Woche noch in meiner Möhre?
- Wie lange hält eine Glühbirne, bevor sie den Geist aufgibt?
- Wie weit rollt ein Apfel vom Stamm?
- Wie weit liegt eine Expertin mit ihrer Schätzung daneben?
- ...

Solche Fragen kann man mit dieser Verteilung darstellen:



Voilà: Die Exponentialverteilung

Darf ich vorstellen: Die Exponential-Verteilung



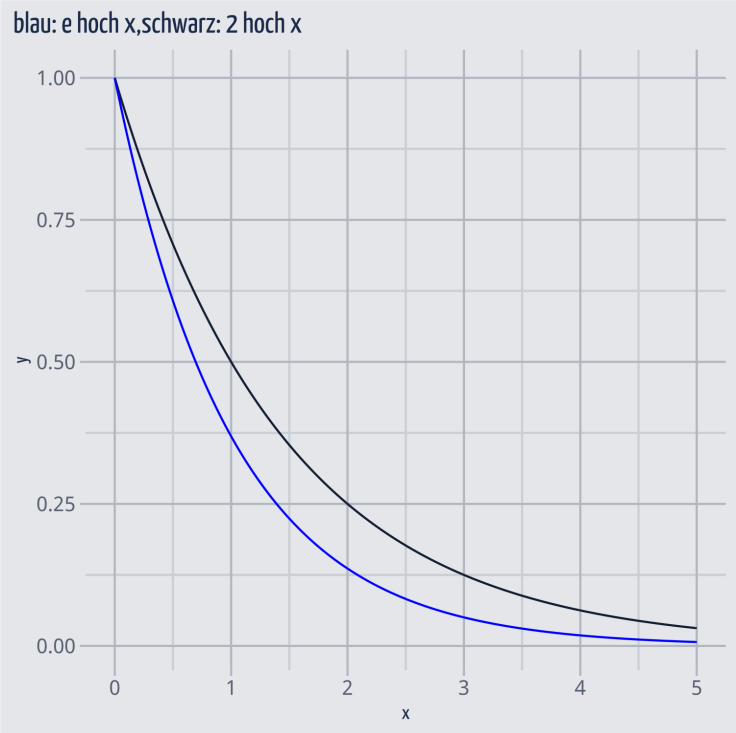
$$X \sim \text{Exp}(1)$$

- Eine *Exp*onentialverteilung ist nur für positive Werte, $x > 0$, definiert.
- Steigt X um eine Einheit, so ändert sich Y um einen konstanten Faktor.
- Sie hat nur einen Parameter, genannt *Rate* oder λ ("lambda").
- $\frac{1}{\lambda}$ gibt gleichzeitig Mittelwert und Streuung ("Gestrecktheit") der Verteilung an.
- Je größer die Rate λ, desto *kleiner* die Streuung und der Mittelwert der Verteilung.
- Je größer $1/\lambda$, desto *größer* die Streuung und der Mittelwert der Verteilung.

Exponentialverteilung berechnen

Im einfachsten Fall gilt: $y = 2^{-x}$ bzw. $y = e^{-x}$ mit $e = 2.71...$

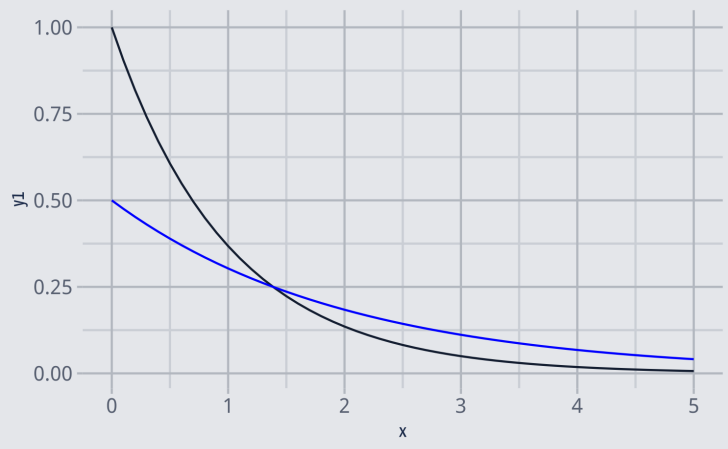
```
d <- tibble(
  x = seq(0, 5, by = 0.01),
  y = 2^(-x),
  y2 = 2.71^(-x))   # e=2.71...

d %>%
  ggplot(aes(x)) +
  geom_line(aes(y=y)) +
  geom_line(aes(y=y2),
            color ="blue") +
  labs(title = paste0("blau: e ho
```



Exponentialverteilung mit R

Für e^x -- Exponenziern mit e (Eulersche Zahl) als Basis -- gibt's in R die Funktion `exp()`. Mit `dexp()` bekommt man die zugehörige Wahrscheinlichkeitsdichte.

```
d <-
  tibble(
    x = seq(0, 5,.1),
    y1 = dexp(x, rate = 1),
    y2 = dexp(x, rate = 0.5)
  )

d %>%
  ggplot(aes(x)) +
  geom_line(aes(y = y1)) +
  geom_line(aes(y = y2),
            color = "blue")
```



$$\beta \sim \text{Exp}(1)$$

$$\beta \sim \text{Exp}(0.5)$$

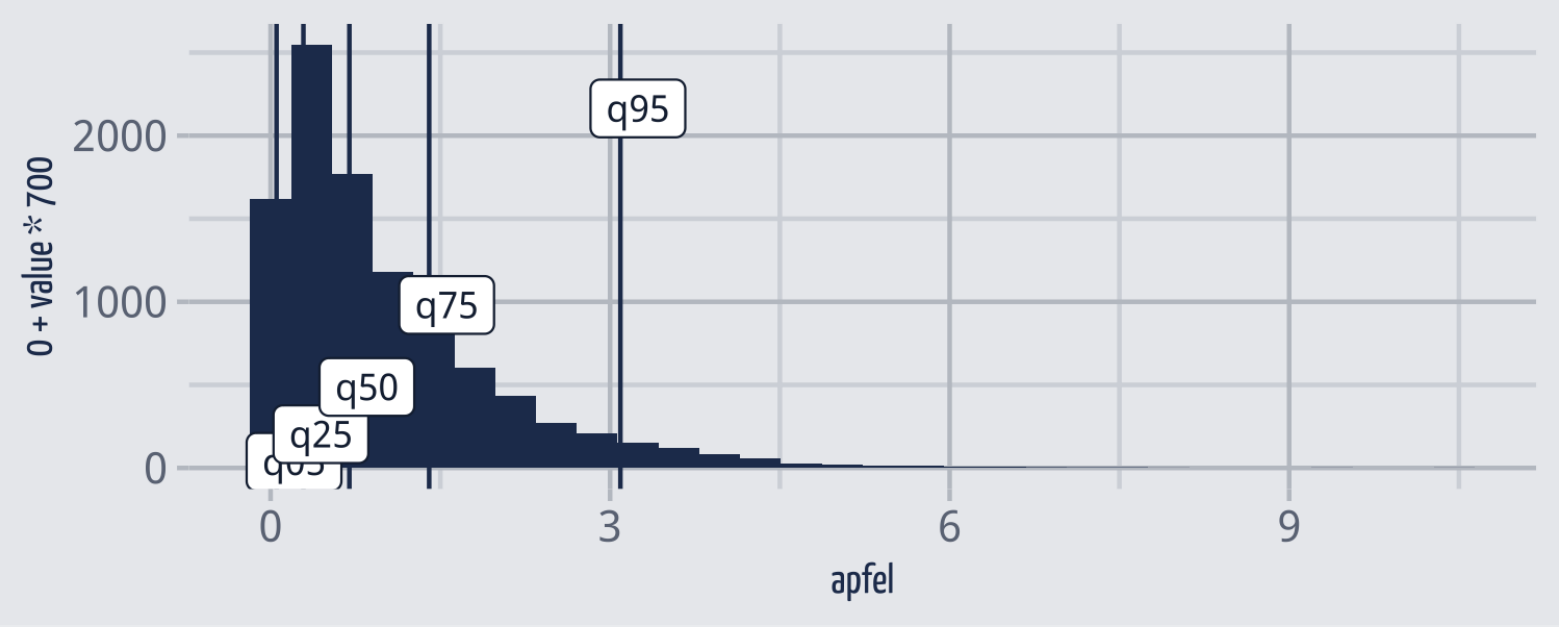
Je kleiner die Rate λ, desto *größer* die Streuung der Verteilung.

Quantile der Exponentialverteilung

... Wenn du nicht mehr weiter weißt, ziehe ein Stichprobe.

Wie weit fällt ein Apfel 🍎 vom Stamm 🌳, wenn wir $\text{Apfel} \sim \mathcal{E}(1)$ annehmen?

```
d <- tibble(apfel = rexp(n = 1e4, rate = 1))

d %>% ggplot(aes(x = apfel)) + geom_histogram()
```

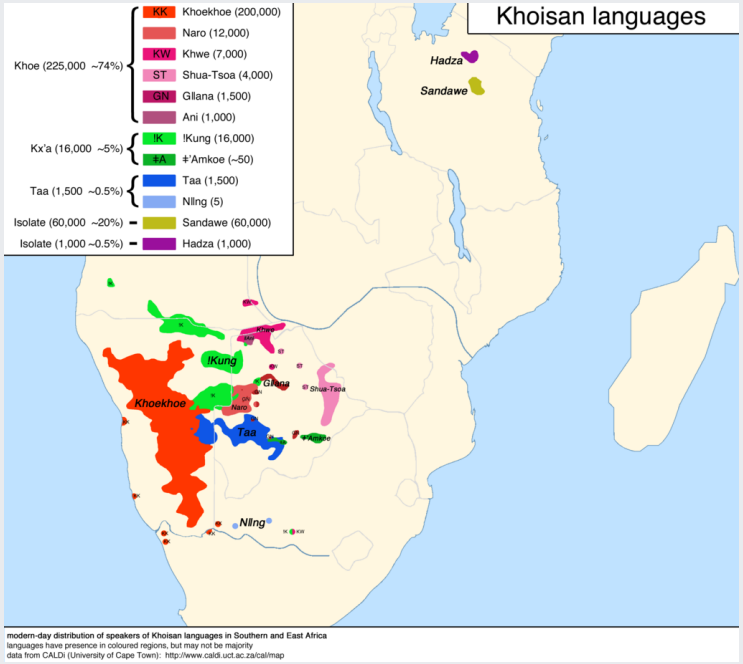


Gaussmodelle

Wie groß sind die !Kung San?

!Kung San

Quelle Internet Archive Book Images, No restrictions, via Wikimedia Commons



By Andrewwik.0 - Own work, CC BY-SA 4.0, Quelle

!Kung Data

Datenquelle

```
library(tidyverse)
Kung_path <-  # Datenquelle s.o.
  "https://tinyurl.com/jr7ckxxj"

d <- read_csv(Kung_path)

d2 <-
  d %>%
  filter(age > 18)
```

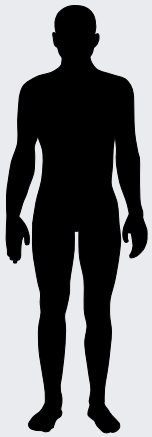
Wir interessieren uns für die Größe der erwachsenen !Kung, $N = 346$:

```
d2 <- d %>%
  filter(age >= 18)
```

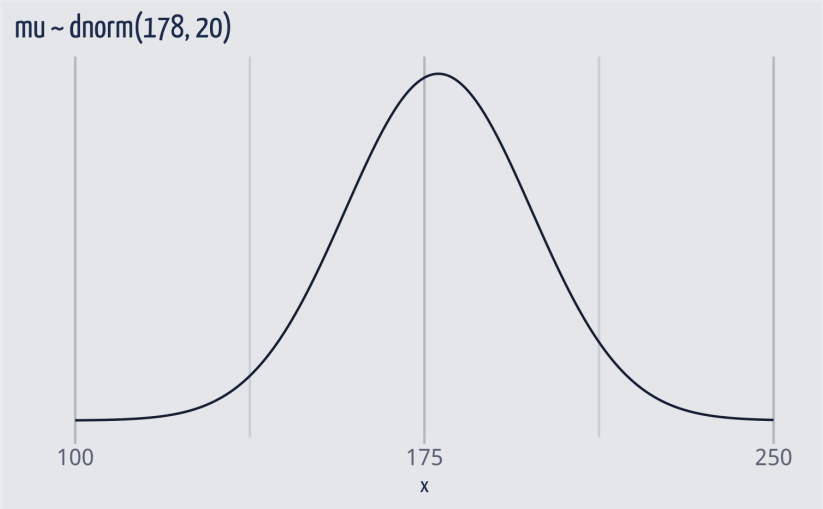
```
library(rstatix)
get_summary_stats(d2)
```

variable	n	min	max	median	q1	q3	iqr	mad	mean	sd	se	ci
age	346	19.00	88.00	40.00	29.00	51.00	22.00	16.31	41.54	15.81	0.85	1.67
height	346	136.53	179.07	154.31	148.59	160.66	12.06	8.47	154.64	7.77	0.42	0.82
male	346	0.00	1.00	0.00	0.00	1.00	1.00	0.00	0.47	0.50	0.03	0.05
weight	346	31.52	62.99	45.01	40.33	49.38	9.04	6.72	45.05	6.46	0.35	0.68

Wir gehen apriori von normalverteilter Größe aus



$$\mu \sim \mathcal{N}(178, 20)$$



Unser Gauss-Modell der !Kung

Wir nehmen an, dass μ und h_i normalverteilt sind und σ exponentialverteilt (da notwendig positiv) ist:

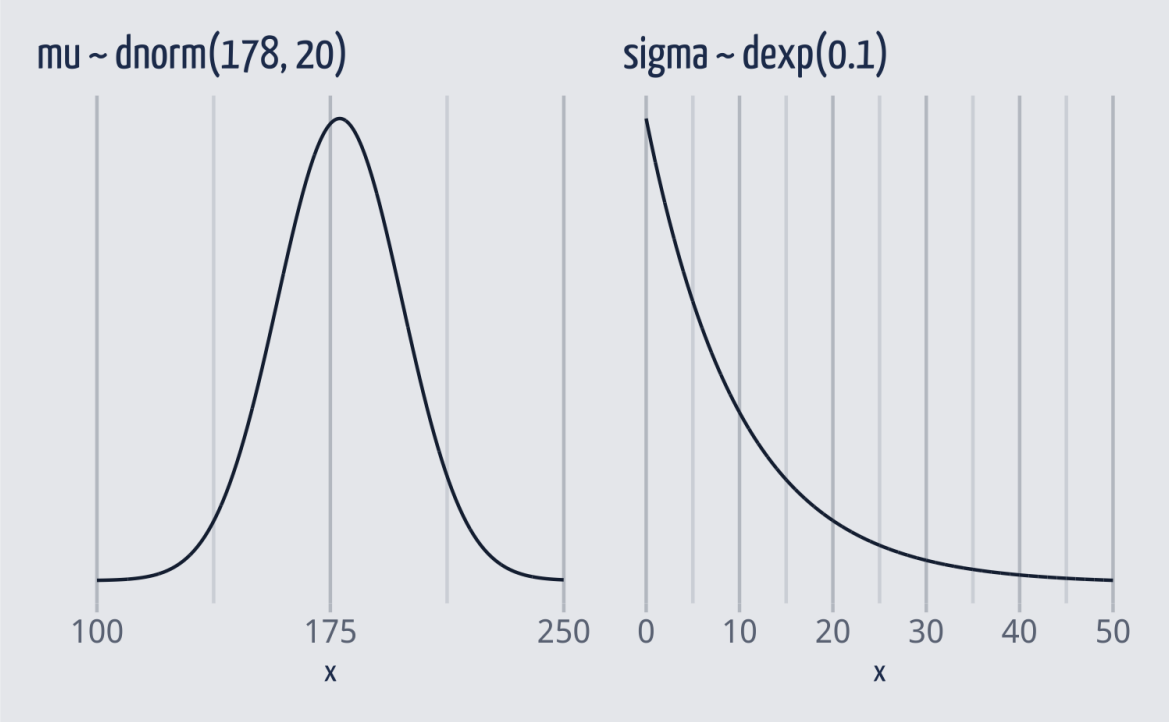
Likelihood:
$h_i \sim \mathcal{N}(\mu, \sigma)$

Prior für μ:
$\mu \sim \mathcal{N}(178, 20)$

Prior für σ:
$\sigma \sim \mathcal{E}(0, 0.1)$

$95\% KI(\mu)$:

178 ± 40



Der Likelihood wird von den Prioris gespeist

Likelihood

Die einzelnen Größen h_i sind normalverteilt mit Mittelwert μ und Streuung σ:

$$h_i \sim \mathcal{N}(\mu, \sigma)$$

Prioris

Mittelwert der Größe ist normalverteilt mit $\mu = 178$ und $\sigma = 20$:

$$\mu \sim \mathcal{N}(178, 20)$$

Die Streuung σ der Größen ist exponentialverteil mit $\lambda = 0.1$.

$$\sigma \sim \mathcal{E}(0.1)$$

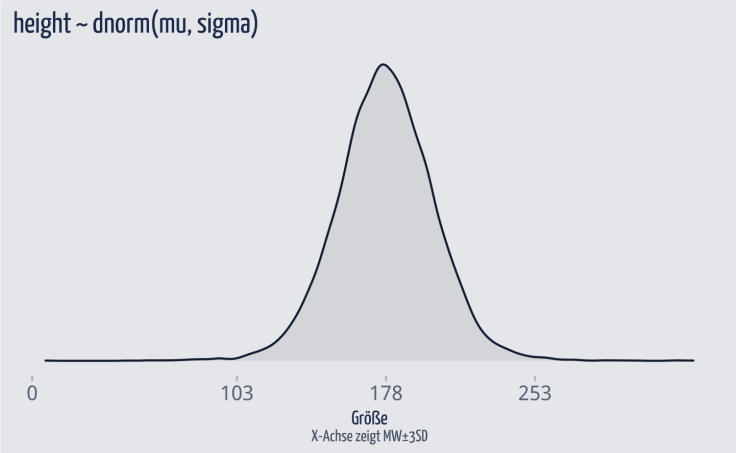
Welche Beobachtungen sind auf Basis unseres Modells zu erwarten?

```
n <- 1e4

sim <- tibble(sample_mu  =
      rnorm(n,
            mean = 178,
            sd   = 20),
    sample_sigma =
      rexp(n,
            rate = 0.1)) %>%
  mutate(height  =
      rnorm(n,
            mean = sample_mu,
            sd   = sample_sigma))

height_sim_sd <-
  sd(sim$height) %>% round()
height_sim_mean <-
  mean(sim$height) %>% round()
```

Quellcode

💭 Was denkt der Golem (`m41`) apriori von der Größe der !Kung?

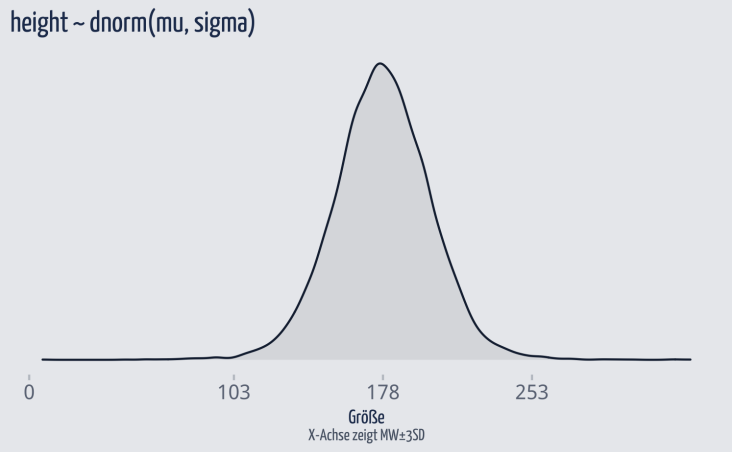
💪 Ziehen wir mal ein paar Stichproben auf Basis des Modells. Voilà:



Priori-Werte prüfen mit der Priori-Prädiktiv-Verteilung

- Die Priori-Prädiktiv-Verteilung (`sim`) simuliert Beobachtungen (nur) auf Basis der Priori-Annahmen: $h_i \sim \mathcal{N}(\mu, \sigma), \mu \sim \mathcal{N}(178, 20), \sigma \sim \mathcal{E}(0.1)$
- So können wir prüfen, ob die Priori-Werte vernünftig sind.

Die Priori-Prädiktiv-Verteilung zeigt, dass unsere Priori-Werte ziemlich vage sind, also einen zu breiten Bereich an Größenwerten zulassen:



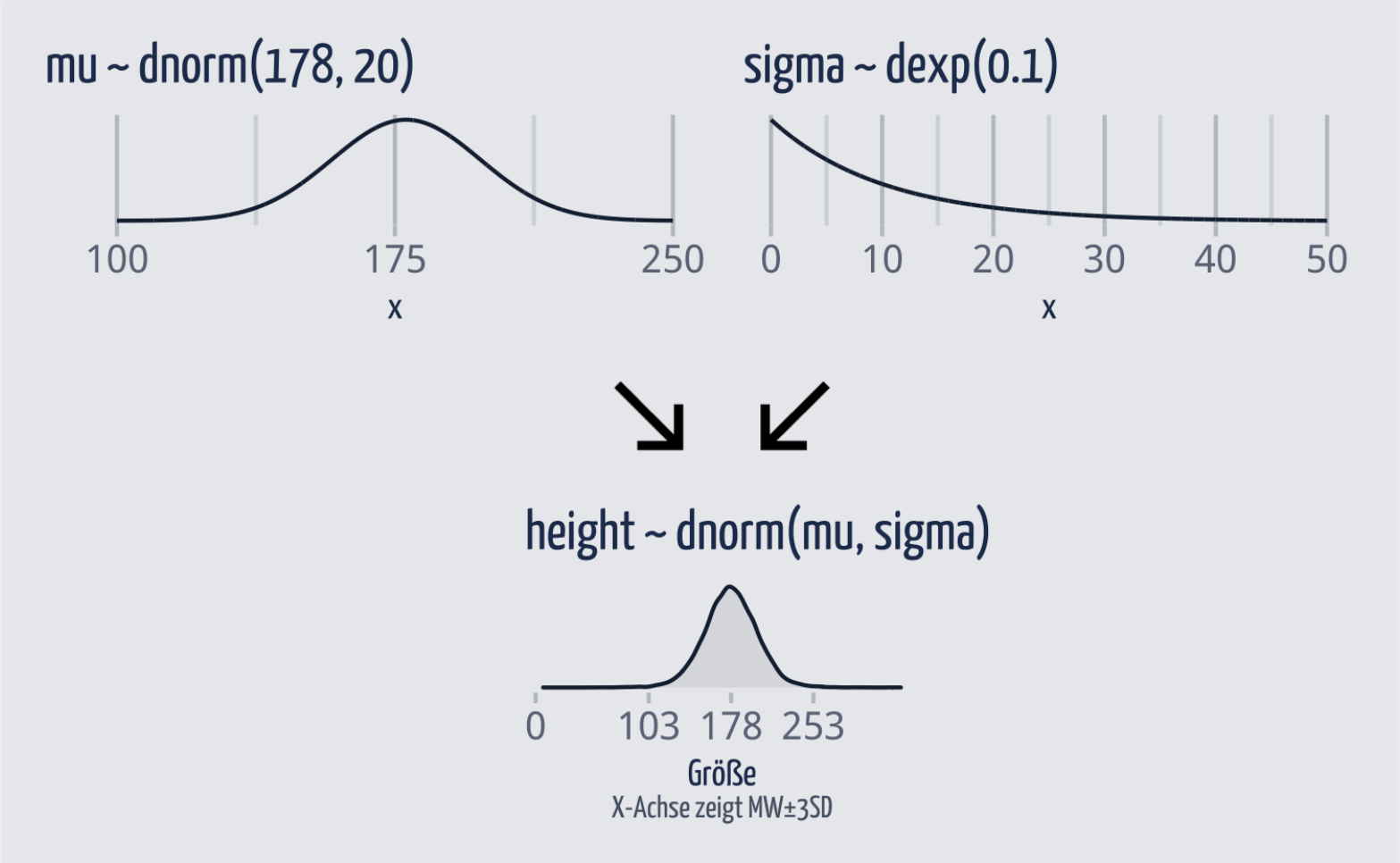
Anteil $h_i > 200$:

```
anteil_großer_kung <-
sim %>%
  count( height > 200) %>%
  mutate(prop = n/sum(n))
anteil_großer_kung
```

```
## # A tibble: 2 × 3
##   `height > 200`     n  prop
##   <lgl>          <int> <dbl>
## 1 FALSE           8328 0.833
## 2 TRUE            1672 0.167
```

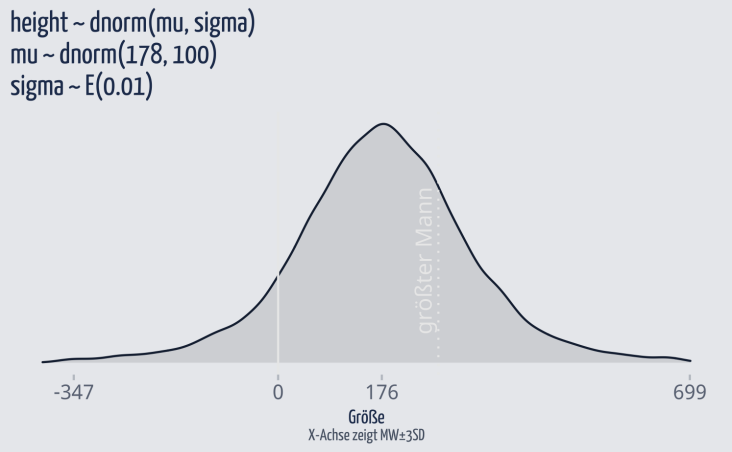
🤔 Sehr große Buschleute? 17 Prozent sind größer als 2 Meter. Das ist diskutabel, muss aber kein schlechter Prior sein.

Vorhersagen der Priori-Werte

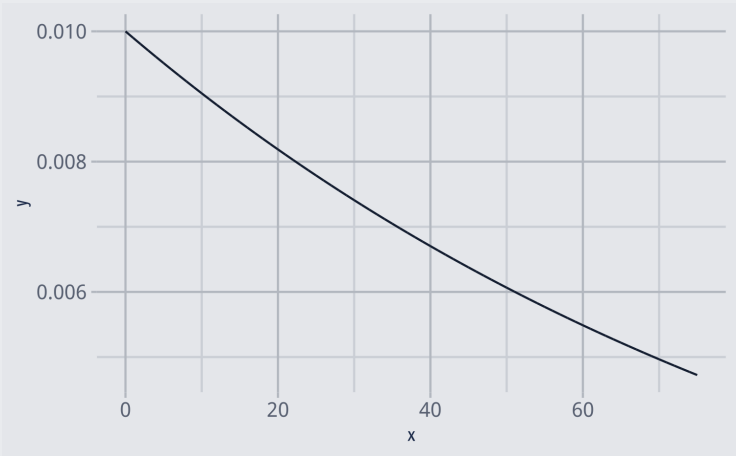


Extrem vage Priori-Verteilung für die Streuung?

$$\sigma \sim \mathcal{E}(\lambda = 0.01)$$



Die Streuung der Größen ist weit:

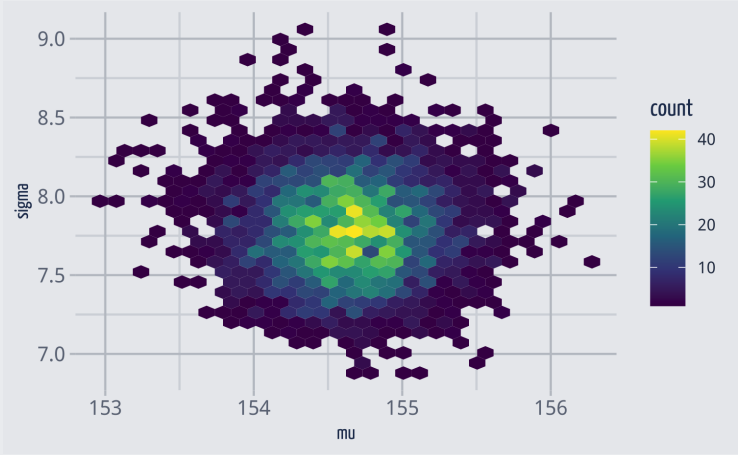


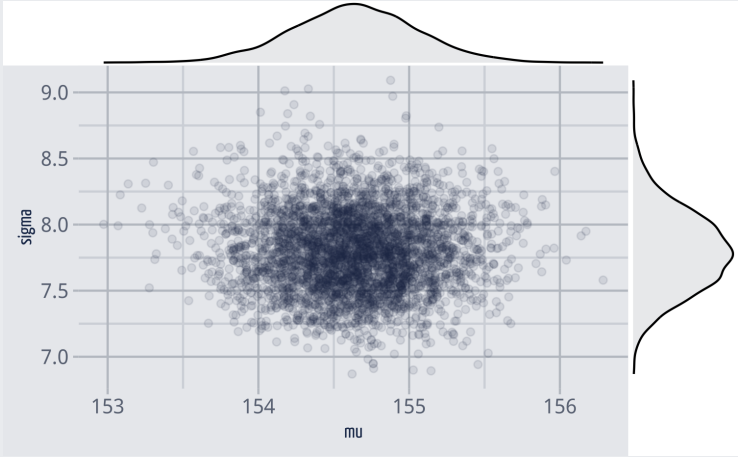
🤔 Das Modell geht apriori von ein paar Prozent Menschen mit *negativer* Größe aus. Ein Haufen Riesen 👹 werden auch erwartet.

🤯 Vage (flache, informationsarme, "neutrale", "objektive") Priori-Werte machen oft keinen Sinn.

Zufällige Motivationsseite

Posteriori-Verteilung des Größen-Modells, `m41`





- Wir bekommen eine Wahrscheinlichkeitsverteilung für μ und eine für σ (bzw. eine zweidimensionale Verteilung, für die μ, σ-Paare).
- Trotz des eher vagen Priors ist die Streuung Posteriori-Werte für μ und σ klein: Die große Stichprobe hat die Priori-Werte überstimmt.
- Ziehen wir Stichproben aus der Posteriori-Verteilung, so können wir interessante Fragen stellen.

Hallo, Posteriori-Verteilung

... wir hätten da mal ein paar Fragen an Sie. 🕵️

- Mit welcher Wahrscheinlichkeit ist die mittlere !Kung-Person größer als 1,55m?
- Welche mittlere Körpergröße wird mit 95% Wahrscheinlichkeit nicht überschritten, laut dem Modell?
- In welchem 90%-PI liegt μ vermutlich?
- Mit welcher Unsicherheit σ ist die Schätzung der mittleren Körpergröße behaftet?
- Welcher Wert der mittleren Körpergröße hat die höchste Wahrscheinlichkeit?

Posteriori-Stichproben mit `stan_glm()` berechnen

- Mit `stan_glm()` können wir komfortabel die Posteriori-Verteilung berechnen.
- Die Gittermethode wird nicht verwendet, aber die Ergebnisse sind - in bestimmten Situationen - ähnlich.
- Es werden aber auch viele Stichproben simuliert (sog. MCMC-Methode).
- Gibt man keine Priori-Werte an, so greift die Funktion auf Standardwerte zurück.

```
library(rstanarm)
# berechnet Post.-Vert.:

stan_glm(
  # modelldefinition:
  AV ~ UV,
, # Datensatz:
  data = meine_daten
)
```

Modelldefinition:

$h_i \sim \mathcal{N}(\mu, \sigma)$, Likelihood

$\mu \sim \mathcal{N}(155, 19)$, Prior Größenmittelwert

$\sigma \sim \mathcal{E}(0.13)$, Prior Streuung der Größen

Ausgabe von `stan_glm()`

```
m41 <- stan_glm(height ~ 1, data = d2)

print(m41)
```

```
## stan_glm
##  family:       gaussian [identity]
##  formula:      height ~ 1
##  observations: 346
##  predictors:   1
## ------
##             Median MAD_SD
## (Intercept) 154.6    0.4
##
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 7.8    0.3
##
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

Wie tickt `stan_glm()`?

Quelle, (Team, 2021)

- *Stan* ist eine Software zur Berechnung von Bayesmodellen; das Paket `rstanarm` stellt Stan für uns bereit.
- `stan_glm()` ist für die Berechnung von Regressionsmodellen ausgelegt.
- Will man nur die Verteilung einer Variablen (wie `heights`) schätzen, so hat man man ... eine Regression ohne Prädiktor.
- Eine Regression ohne Prädiktor schreibt man auf Errisch so: `y ~ 1`. Die `1` steht also für die nicht vorhandene UV; `y` meint die AV (`height`).
- `MAD_SD` ist eine robuste Version der Streuung, mit inhaltlich gleicher Aussage
- `(Intercept)` (Achsenabschnitt) gibt den Mittelwert an.

Dokumentation RstanARM

Stichproben aus der Posteriori-Verteilung ziehen

```
post_m41 <- as_tibble(m41)
print(post_m41)
```

Hier die ersten paar Zeilen von `post_m41`:

(Intercept)	sigma
154.9014	7.951013
154.9072	8.072041
155.0016	7.523712
154.1477	7.599040
154.3553	7.690318
154.7677	7.683235

Mit welcher Wahrscheinlichkeit ist $\mu > 155$?

```
names(post_m41) <-
  c("mu", "sigma")

post_m41 %>%
  count(mu > 155) %>%
  mutate(prop = n/sum(n))
```

```
## # A tibble: 2 × 3
##   `mu > 155`     n  prop
##   <lgl>      <int> <dbl>
## 1 FALSE       3224 0.806
## 2 TRUE         776 0.194
```

Antworten von der Posteriori-Verteilung

Welche mittlere Körpergröße wird mit 95% Wahrscheinlichkeit nicht überschritten, laut dem Modell `m41`?

```
post_m41 %>%
  summarise(
    q95 =
      quantile(mu, .95))
```

```
## # A tibble: 1 × 1
##     q95
##   <dbl>
## 1  155.
```

In welchem 90%-PI liegt μ vermutlich?

```
post_m41 %>%
  summarise(
    pi_90 =
      quantile(mu, c(0.05,
                     0.95)))
```

```
## # A tibble: 2 × 1
##   pi_90
##   <dbl>
## 1  154.
## 2  155.
```

🏋️ Ähnliche Fragen bleiben als Übung für die Leseris 🤓.

Standard-Prioriwerte bei `stan_glm()` 1/3

```
prior_summary(m41)
```

```
## Priors for model 'm41'
## ------
## Intercept (after predictors centered)
##   Specified prior:
##     ~ normal(location = 155, scale = 2.5)
##   Adjusted prior:
##     ~ normal(location = 155, scale = 19)
##
## Auxiliary (sigma)
##   Specified prior:
##     ~ exponential(rate = 1)
##   Adjusted prior:
##     ~ exponential(rate = 0.13)
## ------
## See help('prior_summary.stanreg') for more details
```

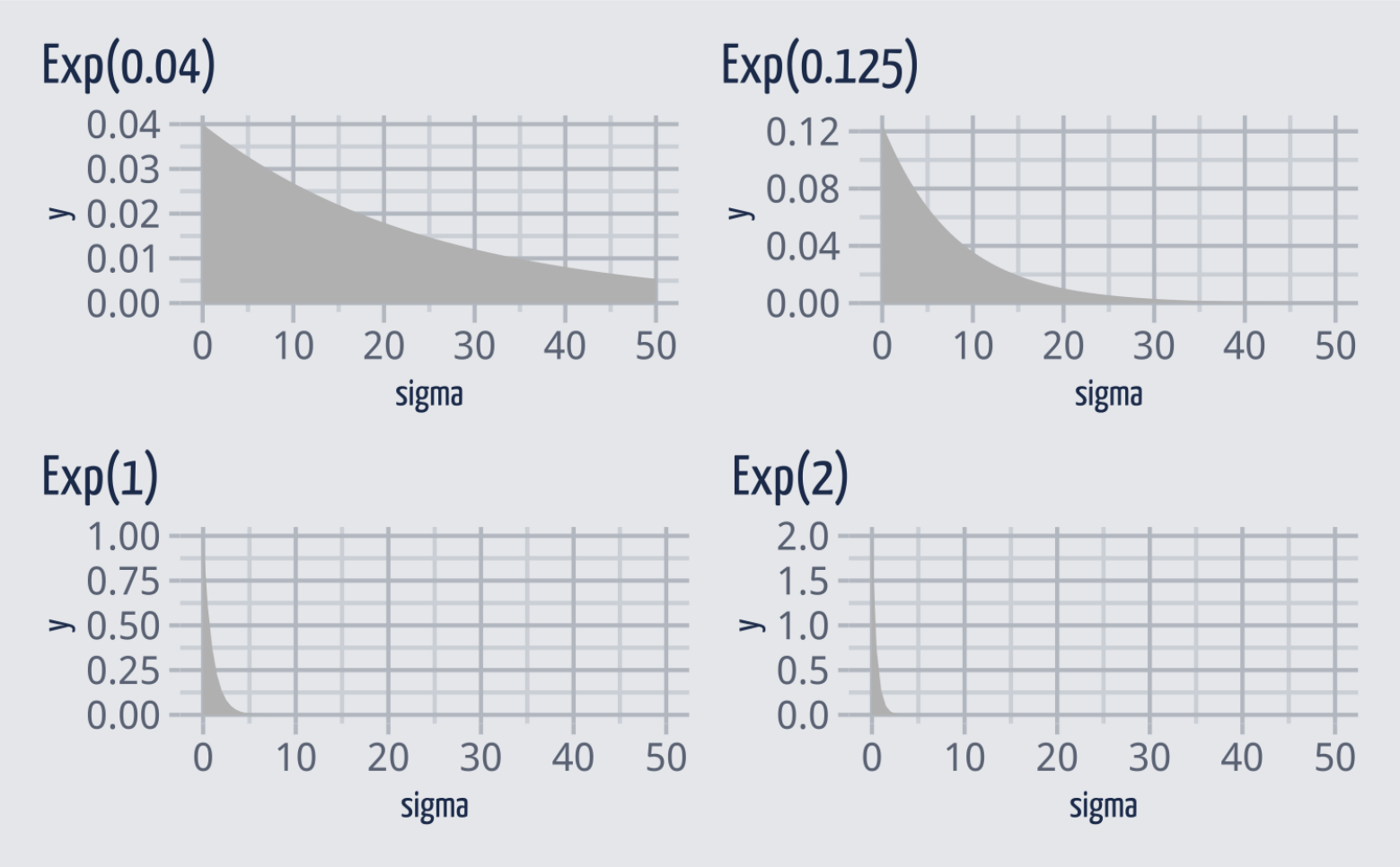
Standard-Prioriwerte bei `stan_glm()` 2/3

- `stan_glm()` verwendet (in der Voreinstellung) *schwach informative* Priori-Werte, die nur wenig Vorabwissen in das Modell geben.
- Es werden dafür die Stichproben-Daten als Priori-Daten verwendet.
- Man sollte diese Standardwerte als Minimalvorschlag sehen. Kennt man sich im Sachgebiet aus, kann man meist bessere Prioris finden.
- Die Voreinstellung hat keinen tiefen Hintergrund; andere Werte wären auch denkbar.

- `Intercept`: μ, der Mittelwert der Verteilung X
  - $\mu \sim \mathcal{N}(\bar{X}, sd(X) \cdot 2.5)$
  - als Streuung von μ wird die 2.5-fache Streuung der Stichprobe angenommen.

- `Auxiliary (sigma)`: σ, die Streuung der Verteilung X
  - $\sigma \sim \mathcal{E}(\lambda = 1/sd(X))$
  - als Streuung von h_i wird 7.8 angenommen.

Visualisierung verschiedener Exponentialverteilungen



Modell m42: unsere Priori-Werte

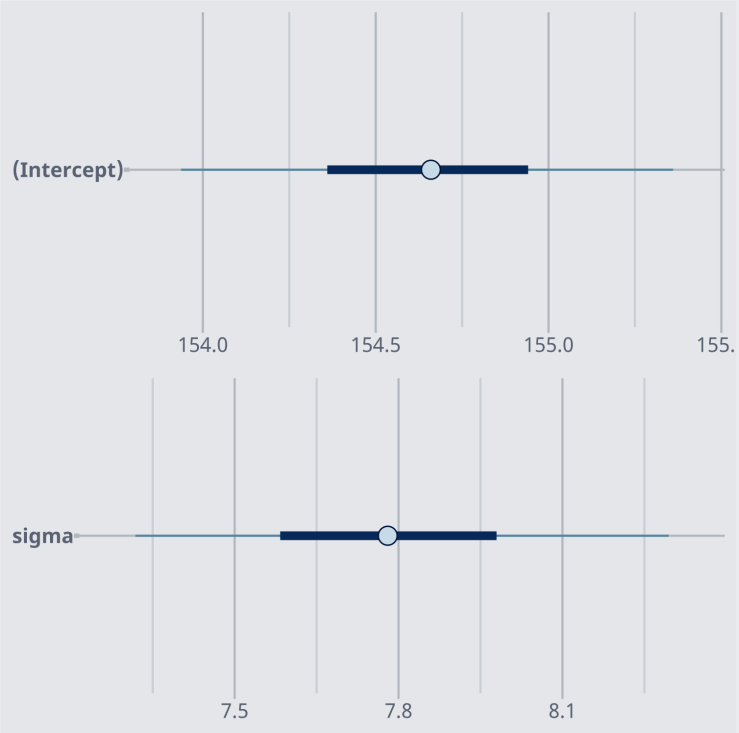
```
m42 <-
  stan_glm(height ~ 1,
           prior_intercept = normal(178, 20),  # mu
           prior_aux = exponential(0.1),  # sigma
           refresh = FALSE,  # bitte nicht so viel Ausgabe drucken
           data = d2)
print(m42)
```

```
## stan_glm
##  family:       gaussian [identity]
##  formula:      height ~ 1
##  observations: 346
##  predictors:   1
## ------
##             Median MAD_SD
## (Intercept) 154.7    0.4
##
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 7.8    0.3
##
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

Posteriori-Verteilung plotten

```
library(bayesplot)
plot(m42,
     pars = "(Intercept)")

plot(m42,
     pars = "sigma")

#plot(m42)
```

Im Standard werden Mediane und 50%- sowie 90%-Perzentilintervalle gezeigt, s. Doku.



Fazit

- Wir haben die Posteriori-Verteilung für ein Gauss-Modell berechnet.
- Dabei hatten wir ein einfaches Modell mit metrischer Zielvariablen, ohne Prädiktoren, betrachtet.
- Die Zielvariable, Körpergröße (`height`), haben wir als normalverteilt mit den Parametern μ und σ angenommen.
- Für μ und σ haben wir jeweils keinen einzelnen (fixen) Wert angenommen, sondern eine Wahrscheinlichkeitsverteilung, der mit der Priori-Verteilung für μ bzw. σ festgelegt ist.

🧡 Bleiben Sie dran!

Hinweise

Zu diesem Skript

- Dieses Skript bezieht sich auf folgende Lehrbücher:
  - Statistical Rethinking, Kapitel 4.1 - 4.3
- Dieses Skript wurde erstellt am 2021-11-08 14:08:29
- Lizenz: CC-BY
- Autor ist Sebastian Sauer.
- Um diese HTML-Folien korrekt darzustellen, ist eine Internet-Verbindung nötig.
- Mit der Taste `?` bekommt man eine Hilfe über Shortcuts.
- Wenn Sie die Endung `.html` in der URL mit `.pdf` ersetzen, bekommen Sie die PDF-Version der Datei. Wenn Sie mit `.Rmd` ersetzen, den Quellcode.
- Eine PDF-Version kann erzeugt werden, indem man im Chrome-Browser druckt (Drucken als PDF).

Literatur

Freeman, M. (2006). "A visual comparison of normal and paranormal distributions". In: *Journal of Epidemiology and Community Health* 60.1, p. 6.

Gelman, A., J. Hill, and A. Vehtari (2021). *Regression and other stories*. Analytical methods for social research. Cambridge University Press.

McElreath, R. (2020). *Statistical rethinking: a Bayesian course with examples in R and Stan*. 2nd ed. CRC texts in statistical science. Taylor and Francis, CRC Press.

Taleb, N. N. (2019). *The statistical consequences of fat tails, papers and commentaries*.

Team, S. D. (2021). *Stan Modeling Language Users Guide and Reference Manual Version 2.28*.