Die Normalverteilung

Jonathan Harrington / Ulrich Reubold

14. May 2019

1. Der Populationsmittelwert: μ

Erstes Beispiel

100 Stück Papier, numeriert mit 0, 1, 2, ...99, liegen in einer Tombolatrommel. Wir ziehen 10 Stück und berechnen den Mittelwert dieser 10 Zufallszahlen. Was ist der *wahrscheinlichste* Mittelwert dieser Zahlen?

Dieser theoretischer Mittelwert, μ, (12ter Buchstabe des griechischen Alphabets, /my:/; wir schreiben im Folgenden "mu") ist der sogenannte Populations-Mittelwert. Für den obigen Fall ist μ:

```
mean(0:99)
```

```
## [1] 49.5
```

Doch warum?

Denken wir ein bisschen nach. Wie oft können wir maximal ziehen (davon ausgehend, dass wir die Papierstücke *nicht* zurücklegen)?

```
losnummern = sample(0:99)
losnummern
```

```
##   [1] 99 22 85 51 17 95  8  1 50 69 66 72 40 73 16 37 20 70 58 23 18 28 24
##  [24] 36 78 62 25 86 57 71 87 83 39  9 33 75 65 12 44 34 26 52 29 54 89  0
##  [47] 56 74 98  2 61 48 92 59 64 82 42 30 90 15 49 27 46 41 79  7 77 88 19
##  [70] 53  4 55 68 43 76 13 97  3 14 67 94 21 10 31 38 63 11 45 32 35 60 81
##  [93] 96  5 93 91 80 47  6 84
```

```
length(losnummern)
```

```
## [1] 100
```

Triviale Antwort: Wir können maximal 100 mal ziehen, denn wir haben nur 100 Lose. Wenn wir die Frage also abändern und fragen, was der Mittelwert sei, wenn wir alle 100 Lose, die mit 0...99 numeriert sind, ziehen, ergibt sich die ebenso triviale Lösung (denn die Reihenfolge der Zahlen spielt beim Bilden des Mittelwertes keinerlei Rolle):

```
mu=mean(sample(0:99))
mu==mean(0:99)
```

```
## [1] TRUE
```

```r
mean(sample(0:99))
```

```
## [1] 49.5
```

Der 'wahre' Mittelwert der gesamten Daten ist also einfach `mean(0:99)=49.5`; um diesen Populationsmittelwert μ zu bestimmen, kann man das entweder

- bestimmen, indem man 100 mal einen Zettel zieht

oder (weil wir in einer sehr kleinen, speziellen Welt leben, und wissen, dass es nur die Zahlen 0 bis 99 gibt, die jeweils 1 mal in der Trommel liegt)

- einfach `mean(0:99)` rechnet.

Wenn eine Durchschnittsnote errechnet wird, macht man ja genau das: man ermittelt alle Noten, schreibt diese auf und errechnet dann das arithmetische Mittel. Das ist machbar, denn die Noten müssen ohnehin bestimmt werden, diese zu notieren und den Mittelwert zu errechnen ist dann kein größerer Aufwand mehr.

Schwieriger wird dies allerdings, wenn wir es mit einer sehr großen Population zu tun haben. Stellen Sie sich vor, Sie wollen die Durchschnittskörpergröße aller in Deutschland lebender Männer im Alter zwischen 18 und 90 Jahren ermitteln. Wir wissen die Körpergrößen der einzelnen Männer nicht (im Gegensatz zu den Zahlen auf den 100 Losen), können also nichts errechnen, solange wir nicht die Körpergrößen ermittelt haben. Um den Populationsmittelwert zu ermitteln, müssten wir also von Millionen von Männer, die in Deutschland leben, die Adresse ermitteln, dort mit einem Metermaß hinfahren, die Person auch antreffen, und dann die Größe von jedem einzelnen Mann messen. Erst dann hätten wir die sogenannte **Grundgesamtheit** erfasst. Das ist schon rein aus wirtschaltlichen Gründen unmöglich.

Stattdessen könnte man *Stichproben* erheben, also nur einen Teil der männlichen Bevölkerung ausmessen - und hoffen, dass der so erhaltene Wert nicht allzu weit weg ist vom tatsächlichen Populationsmittelwert μ. Dies ist genau das, was man bei der Erhebung von Daten in der empirischen Wissenschaft macht.

Dabei ist natürlich nicht nur der Mittelwert an sich interessant (was genau sagt es uns, wenn die Geburtenrate bei 1.5 Kindern/Frau liegt?), sondern auch, wie die Daten um diesen (Mittel-) Wert herum verteilt sind. Hierbei sind weniger die Extrema interessant (z.B. "kleinster/größter Mann Deutschlands"), sondern wie sehr die Werte für *die meisten zur Grundgesamtheit gehörenden* Menschen um den Mittelwert schwanken. Wie man an vielen natürlichen Größen feststellen kann, folgen die Verteilungen dieser Größen zumeist einer sogenannten **Normalverteilung**.

Einer solche Normalverteilung können wir uns anhand des obengenannten Beispieles annähern. Wie wir feststellten, liegt der Populationsmittelwert bei 49.5. Wenn wir nur eine Stichprobe ermitteln, indem wir z.B. nur 10 Lose ziehen, liegt der Wert wahrscheinlich nicht bei exakt 49.5, sondern ist mal höher, mal niedriger:

```r
mean(sample(0:99,10))
```

```
## [1] 36.7
```

```r
mean(sample(0:99,10))
```

```
## [1] 64.3
```

```r
mean(sample(0:99,10))
```

```
## [1] 52.4
```

```r
mean(sample(0:99,10))
```

```
## [1] 53.3
```

```r
mean(sample(0:99,10))
```

```
## [1] 38.1
```

```r
mean(sample(0:99,10))
```

```
## [1] 53.4
```

```r
mean(sample(0:99,10))
```

```
## [1] 43.6
```

```r
mean(sample(0:99,10))
```

```
## [1] 51.9
```

```r
mean(sample(0:99,10))
```

```
## [1] 48.5
```

```r
mean(sample(0:99,10))
```

```
## [1] 51.5
```

Auch wenn wir dies jetzt 10 mal gemacht haben, ist dies nicht das gleiche, als wenn wir 100 Lose gezogen hätten, da wir ja mit jeder der 10 Stichprobenziehungen von neuem - also bei einer vollständig gefüllten Trommel - angefangen haben. Dennoch werden wir wahrscheinlich festellen, dass die ermittelten Werte meistens nicht allzu weit weg sind von $\mu = 49.5$. Es gibt hier Grenzen. Im Extremfall könnten wir zufällig die Zahlen 0, 1, 2, 3, 4, 5, 6, 7, 8, und 9 ziehen, oder - im anderen Extrem die Zahlen 90 bis 99. Unser Stichproben-Mittelwert (wir werden ihn später “m” nennen) kann also schwanken zwischen

```r
mean(0:9)
```

```
## [1] 4.5
```

```r
mean(90:99)
```

```
## [1] 94.5
```

Dass aber diese Werte herauskommen, ist zwar möglich, aber sehr unwahrscheinlich. Wenn ich im obengenannten Lose-Beispiel irgendein Los ziehe, ist die Wahrscheinlichkeit, dass auf dem Los z.B. 99 steht, genau `1/100=0.01`, denn es gibt ja noch 99 andere Möglichkeiten.

Wenn ich 10 Lose ziehe, ist es aber sehr unwahrscheinlich - so sagt uns schon unsere Intuition - dass wir genau eine dieser extremen Zahlenkombinationen ziehe. Viel wahrscheinlicher erscheint es doch, dass wir Zahlen ziehen, die einigermaßen gleichmäßig über den Zahlenraum 0:99 verteilt sein werden - und deren Mittelwert deshalb nicht allzu weit von 49.5 entfernt sein wird.

Um zu zeigen, dass dies tatsächlich so ist, wählen wir ein etwas begrenzteres Beispiel.

Zweites Beispiel

Wir werfen einen Würfel k mal (oder k Würfel gleichzeitig einmal). Wir berechnen den Mittelwert der k Zahlen. Was ist μ?

2. Der Stichprobenmittelwert: m

Wenn wir 10 Würfel werfen, bekommen wir 10 Zufallswerte, die zwischen 1 und 6 schwanken können. In `R` können wir das auch mit `sample()` tun, müssen aber bedenken, dass es (im Gegensatz zur Lostrommel oben) sein kann, dass die gleiche Zahl nochmal kommt. Daher müssen wir `sample(...,replace=TRUE)` aufrufen:

```
# Einen Würfel werfen
sample(1:6, 1, replace=T)

## [1] 5

# 10 Würfel werfen
sample(1:6, 10, replace=T)

##  [1] 4 5 4 6 6 3 2 3 6 2
```

Auch hier gilt: der Populationsmittelwert μ lässt sich einfach errechnen, auch wenn wir unendlich oft würfeln können - wir können uns also anstrengen, wie wir wollen, wir können die Grundgesamtheit nicht erfassen (dennoch bleibt es bei dem Namen Grundgesamtheit - es ist dann eine sogenannte *überabzählbar unendliche Grundgesamtheit*). Es bleibt aber dabei (wie beim Lose-Beispiel), dass jede Zahl die gleiche Vorkommenswahrscheinlichkeit hat - in unserem Fall 1/6, und wir μ daher rechnen können durch:

```
mu = mean(1:6)
mu

## [1] 3.5
```

Den Mittelwert von (im obigen Beispiel 10) Zufallswerten nennen wir den Stichprobenmittelwert und verwenden das Symbol m:

```
m = mean(sample(1:6, 10, replace=T))
m
```

```
## [1] 3.2
```

Auch hier gilt: m und μ werden fast immer voneinander abweichen.

3. Beziehung zwischen Stichproben- und Populationsmittelwert

Wir werfen k (10) Würfel N (20) Mal und berechnen jedes Mal den Mittelwert der 10 Zahlen

```
# Wieviele Würfel werden geworfen?
k = 10
# Wie oft wird geworfen?
N = 20

wuerfel <- NULL
for(j in 1:N){
ergebnis = mean(sample(1:6, k, replace=T))
wuerfel = c(wuerfel, ergebnis)
}

wuerfel
```

```
##  [1] 3.7 3.3 2.9 3.3 4.0 3.5 2.4 2.9 3.3 4.1 2.8 3.9 3.3 3.0 2.7 4.6 3.7
## [18] 2.8 2.9 4.0
```

Der Mittelwert dieser 20 Mittelwerte nähert sich an μ an:

```
mean(wuerfel)
```

```
## [1] 3.355
```

```
# Je höher N, umso näher an μ. 20 Würfel werfen. Den Mittelwert
# davon berechnen. 5000 Mal wiederholen. Daher 5000 Mittelwerte
k=20
N = 5000
wuerfel <- NULL
for(j in 1:N){
    ergebnis = mean(sample(1:6, k, replace=T))
    wuerfel = c(wuerfel, ergebnis)
}
# Der Mittelwert dieser 5000 Mittelwerte
mean(wuerfel)
```

```
## [1] 3.49307
```

Nur wenn wir diesen Vorgang unendlich viel Mal wiederholen könnten, dann bekämen wir exakt μ = 3.5 durch `mean(wuerfel)`.

4. Verteilung der Stichprobenmittelwerte; Wahrscheinlichkeitsdichte

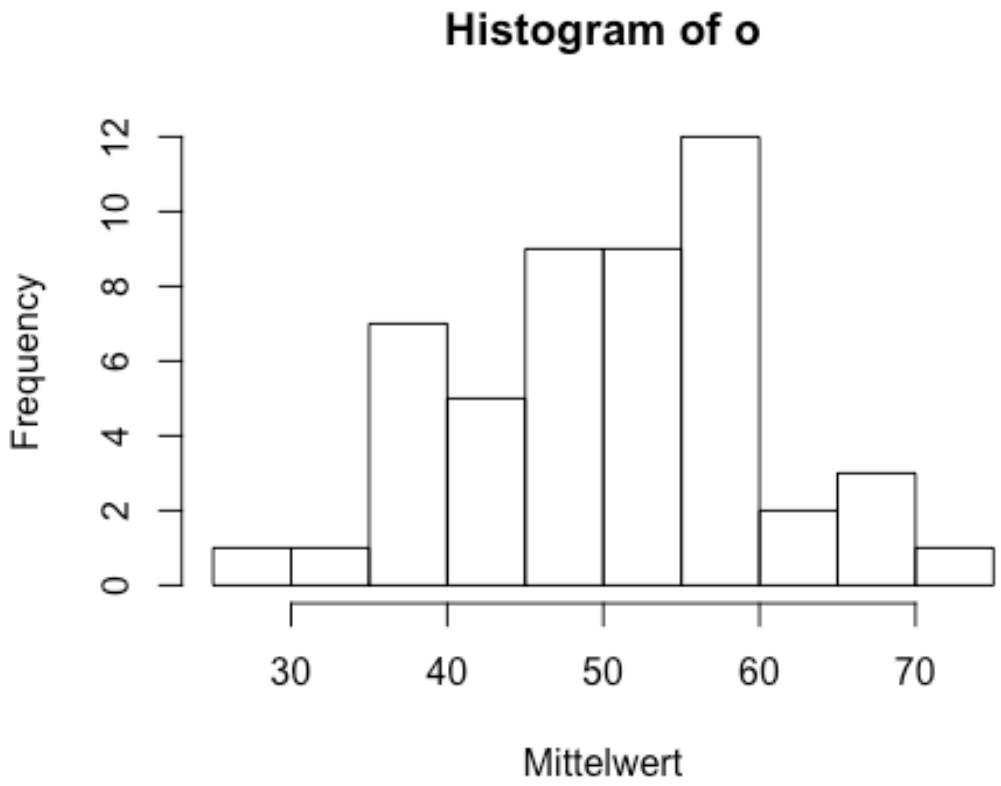
Allgemeine Funktion mit den folgenden Defaults:

Wir werfen k Würfel, berechnen den Mittelwert davon, wiederholen diesen Vorgang 50 mal - und bekommen daher 50 Mittelwerte

```
proben <- function(unten=1, oben = 6, k = 10, N = 50)
{
# default: wir werfen 10 Wuerfel 50 mal
alle <- NULL
for(j in 1:N){
ergebnis = mean(sample(unten:oben, k, replace=T))
alle = c(alle, ergebnis)
}
alle
}

# Funktion durchführen mit den obigen Defaults
proben()
```

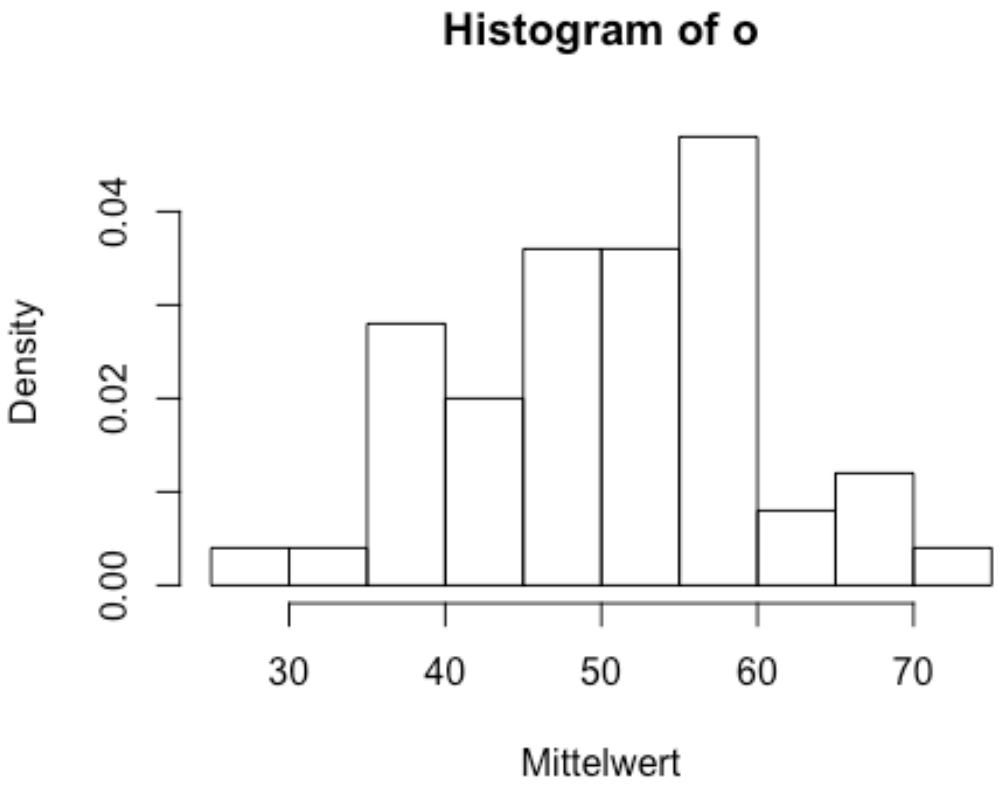
```
##  [1] 3.4 3.5 3.5 3.9 3.3 3.8 3.5 3.1 3.2 3.1 3.7 3.9 4.2 3.7 3.1 3.4 3.7
## [18] 3.6 3.3 3.5 2.9 3.5 3.7 3.2 3.6 4.0 4.0 3.5 4.7 3.3 3.4 2.2 3.9 4.0
## [35] 4.6 2.8 4.5 3.4 3.3 2.7 2.3 3.9 3.6 3.7 2.9 3.2 3.7 3.8 2.7 3.6
```

```
# 100 Stück Papier nummeriert 0, 1, ... 99 in einem Hut.
# Wir ziehen 10 Stück Papier, berechnen den Mittelwert,
# (legen die 10 Stück wieder in den Hut hinein)
# wiederholen diesen Vorgang 50 mal, bekommen 50 Mittelwerte
o = proben(0, 99, 10, 50)
# Ein Histogramm dieser 50 Mittelwerte
hist(o, xlab = "Mittelwert")
```

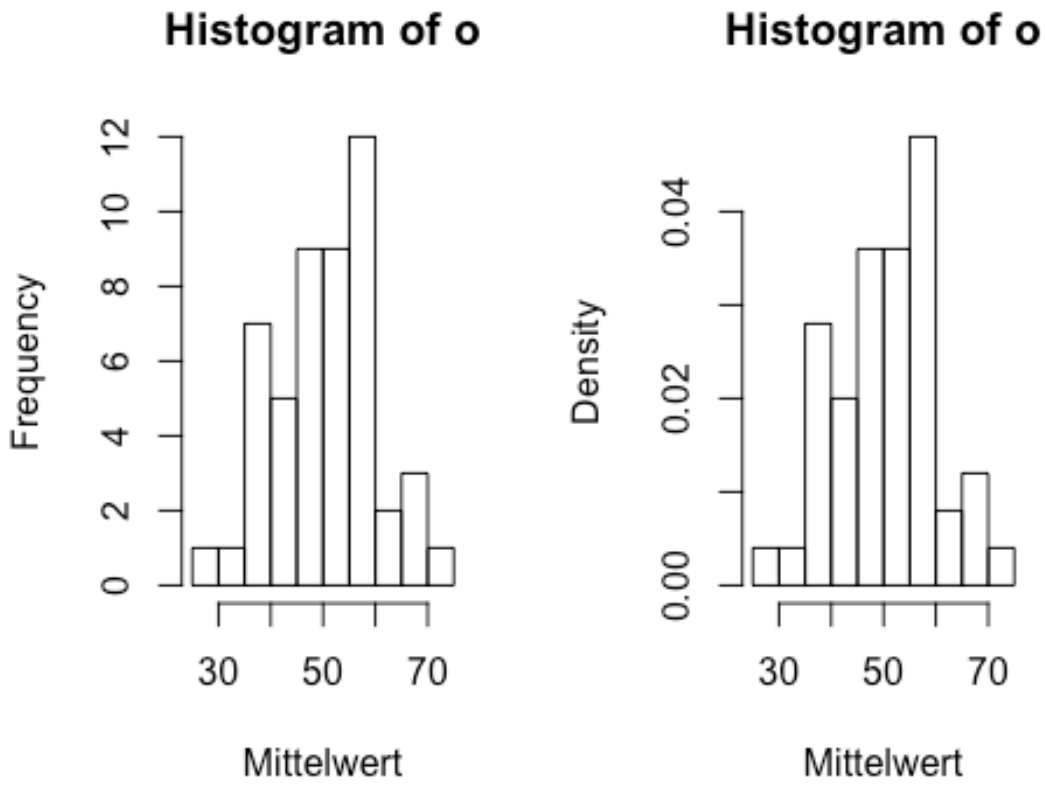


Wahrscheinlichkeitsdichte: Die Wahrscheinlichkeitsdichte ist eine Umsetzung der vertikalen Achse, sodass die Flächensumme der Balken den Wert 1 ergibt.

```
hist(o,  freq=FALSE, xlab = "Mittelwert")
```




```r
# Beide Abbildungen nebeneinander
par(mfrow=c(1,2))
hist(o, xlab = "Mittelwert")
hist(o,  freq=FALSE, xlab = "Mittelwert")
```




```
# Zur Info: wenn das Histogramm aus N Werten besteht, dann:
# Dichte = Balkenhöhe/(Balkenbreite * N)
```

5. Annäherung an die Normalverteilung

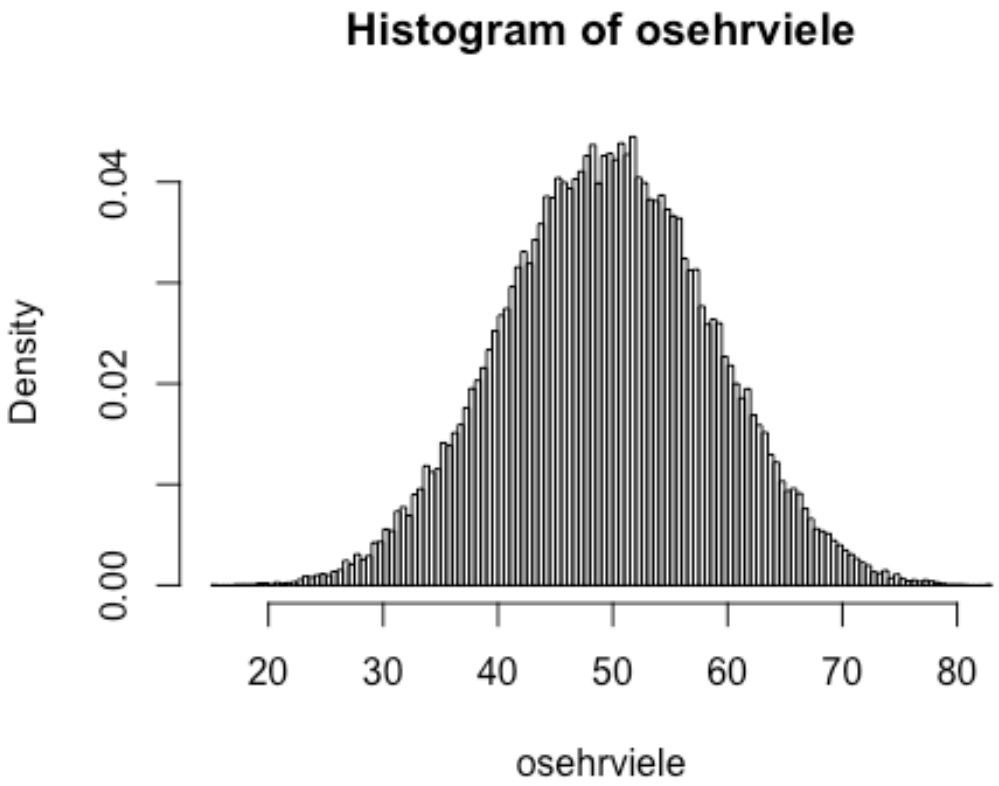
Die Normalverteilung ist ein Histogramm, das unter zwei Bedingungen erstellt wird:

(a) nicht N = 50 sondern unendlich viele Mittelwerte

(b) wir lassen mit zunehmenden Stichproben die Balkenbreite immer kleiner werden, sodass im unendlichen Fall die Balkenbreite unendlich klein (= 0) ist (also wird die Balkenfläche zu einer Linie). Daher haben wir keine Stufen mehr (von einem Balken zum nächsten) sondern eine glatte Kurve.

Beispiel: ein Hut mit 100 Stück Papier; wir ziehen 10 Stück, berechnen den Mittelwert. Wir wiederholen diesen Vorgang 50000 mal (kann länger dauern!):

```
osehrviele = proben(0, 99, 10, 50000)
# Histogramm erstellen mit 200 Intervallen
par(mfrow=c(1,1)) #auf Darstellung nur einer Abbildung zurückstellen
hist(osehrviele, breaks=200, freq=F)
```

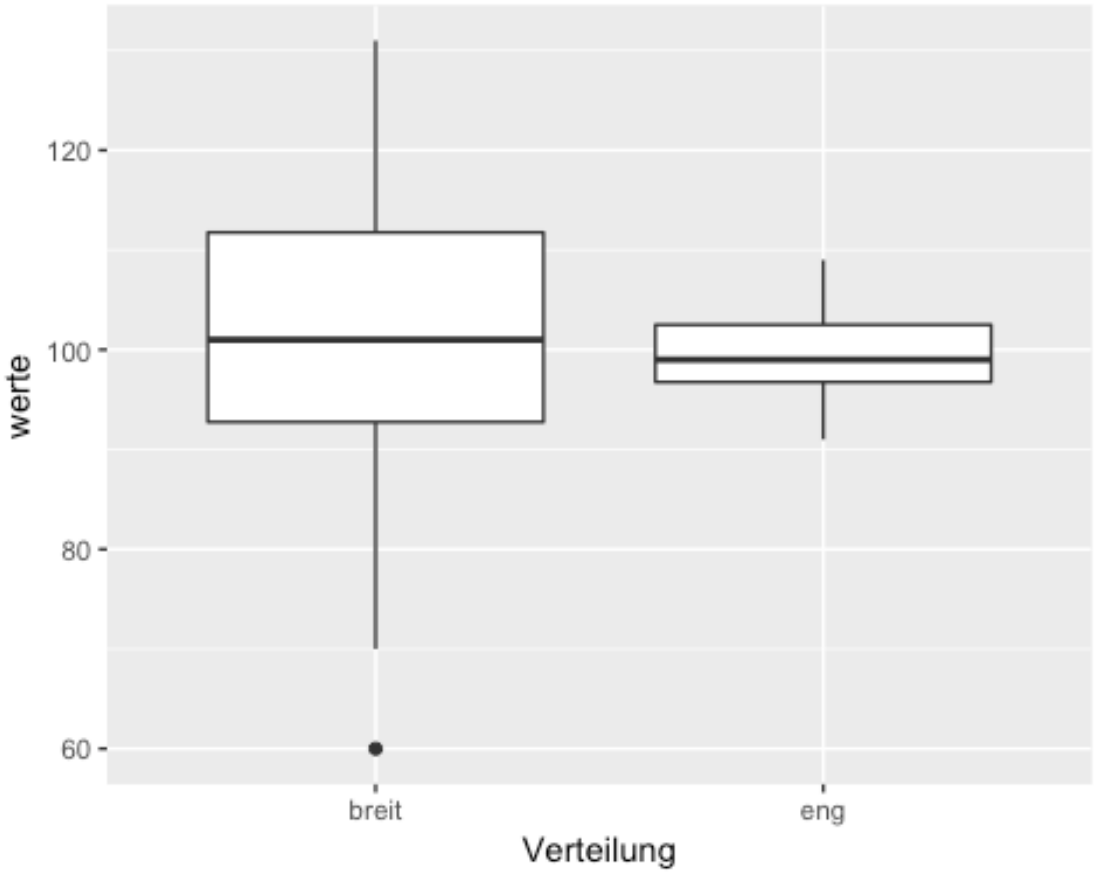


6. Die Standardabweichung

i. Die Stichproben-Standardabweichung: s `sd()`

Die **Standardabweichung** (genauer: die Stichproben-Standardabweichung) s misst die Streuungsgröße einer beliebigen Verteilung um deren Mittelwert, z.B:

```
nex = read.table(file.path(pfadu, "normexample.txt"))

library(ggplot2)
ggplot(nex) +
  aes(y = werte, x = Verteilung) +
  geom_boxplot()
```



Die Standardabweichungen dieser Verteilungen errechnet man mit der `sd()`-Funktion, die man mittels des Packets `dplyr` hier auf die Werte pro Faktorstufe in der Spalte Verteilung anwendet:

```
library(dplyr)

nex %>% #%>% `pipe`-Symbol; nimm das Ergebnis dieser Zeile und schicke es als Input zur nächsten Zeile
  group_by(Verteilung) %>% #mache das Folgende pro Stufen des Faktors "Verteilung"
  summarise(sd=sd(werte)) #erzeuge eine Spalte, die sd heißt, und die die Standardabweichungen der Werte in $werte (wie gesagt, aufgeteilt nach $Verteilung-Stufen) enthalten

## # A tibble: 2 x 2
##   Verteilung    sd
##   <fct>      <dbl>
## 1 breit      17.4
## 2 eng         4.74
```

D.h., je kleiner der Wert der Standardabweichung, desto "enger" ist die Verteilung.

Alternativ hätten wir auch den `aggregate()`-Befehl benutzen können:

```
aggregate(werte~Verteilung,data=nex,FUN=sd)
```

```
##   Verteilung     werte
## 1      breit 17.364210
## 2        eng  4.739531
```

Mit dieser Funktion können wir auch die `summary()`-Funktion anwenden können, um die Werte zu bekommen, der der Boxplot oben zeigt (plus das arithmetische Mittel, aber minus das 95%-Konfidenzintervall (siehe unten)):

```
aggregate(werte~Verteilung,data=nex,FUN=summary)
```

```
##   Verteilung werte.Min. werte.1st Qu. werte.Median werte.Mean
## 1      breit      60.00         92.75       101.00     100.40
## 2        eng      91.00         96.75        99.00      99.60
##   werte.3rd Qu. werte.Max.
## 1        111.75     131.00
## 2        102.50     109.00
```

Die Boxen zeigen 50% der Daten, nämlich alles vom 25%-Quantil bis zum 75%-Quantil. Der dicke Strich ist der Median Wert. Die Striche außerhalb der Box zeigen die Spannweite für 95% der Daten an (das 95%-Konfidenzintervall - siehe unten). Sollte es Werte außerhalb dieses Bereiches geben, werden diese "Ausreißer" als Punkte dargestellt.

ii. Die Populations-Standardabweichung: σ (sigma)

Die **Populations-Standardabweichung σ** ist die theoretische Standardabweichung unendlich vieler Stichproben.

```
# z.B.
#
# Ich werfe einen Würfel 20 mal

#Stichproben-Standardabweichung:
o = proben(k = 1, N = 20)
sd(o)
```

```
## [1] 1.76516
```

```
# Ich werfe den Würfel unendlich viel mal:
# Populationsstandardabweichung, σ (griechisch, `sigma`)
sd(1:6) * sqrt(5/6)
```

```
## [1] 1.707825
```

Annäherung an die Populations-Standardabweichung:

```
# Ich werfe einen Würfel 50000 mal
o = proben(k = 1, N = 50000)
# Stichproben-Standardabweichung (müsste ziemlich nah
# an σ sein)
sd(o)
```

```
## [1] 1.707668
```

iii. Der Standard-Error (*standard error of the mean, SE*)

Der Standard-Error ist die Populations-Standardabweichung von Mittelwerten. Zum Beispiel:

Ich werfe 4 Würfel, berechne den Mittelwert, wiederhole diesen Vorgang unendlich viel mal, bekomme unendlich viele Mittelwerte.

Der Standard-Error (SE) ist die Standard-Abweichung dieser Mittelwerte und ist in diesem theoretischen Fall $\sigma/\sqrt{k}$ (wobei k die Anzahl der Würfel ist).

Für diesen Fall (Mittelwerte von 4 Würfeln) ist der SE:

```
sd(1:6) * sqrt(5/6) / sqrt(4)
```

```
## [1] 0.8539126
```

Weiteres Beispiel:

Hut mit 100 Zahlen nummeriert 0, 1, 2, ..99.

Wir ziehen 10 Zahlen, berechnen den Mittelwert (tun die Zahlen wieder in den Hut hinein), ziehen wieder 10 Zahlen, berechnen den Mittelwert, wiederholen diesen Vorgang unendlich viel mal...

Was ist SE?

```
sd(0:99) * sqrt(99/100) / sqrt(10)
```

```
## [1] 9.128253
```

```
# Wir müssten diesen Wert mit ziemlich vielen Wiederholungen
# annähern
# Von oben:
# osehrviele = proben(0, 99, 10, 50000)
sd(osehrviele)
```

```
## [1] 9.144947
```

7. Überlagerung der theoretischen Normalverteilung auf eine Stichprobe

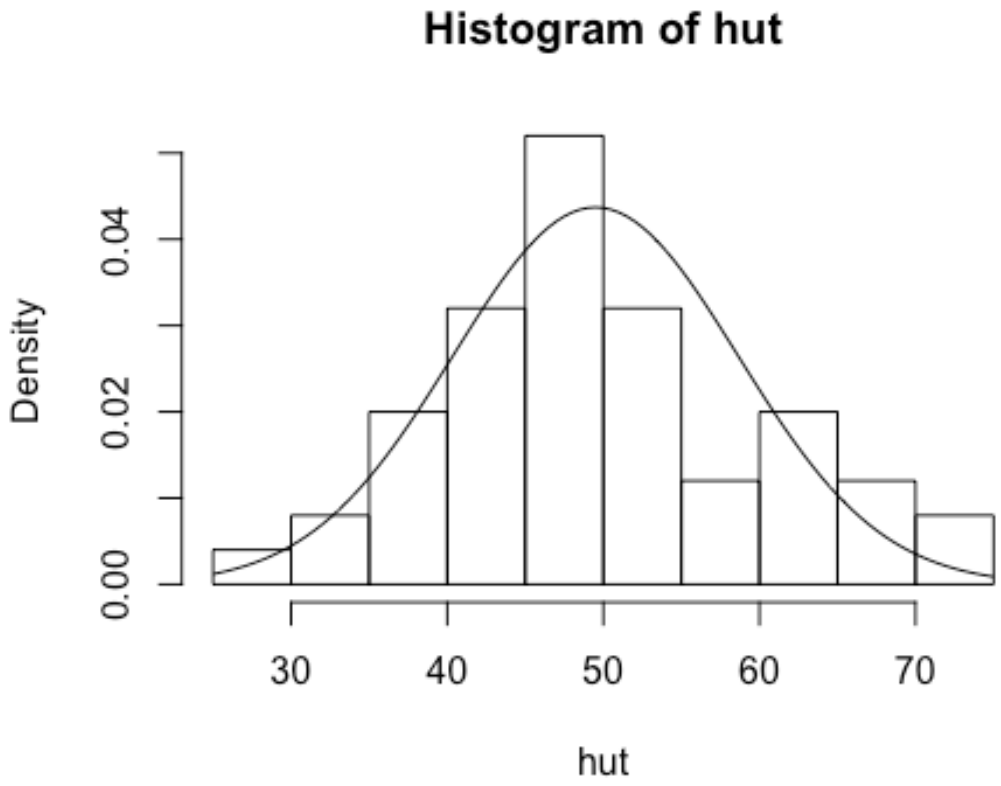
Wenn μ und SE im theoretischen Fall (= unendlich viele Wiederholungen) bekannt sind, dann kann eine theoretische Normalverteilung auf die Stichprobe überlagert werden.

Beispiel Hut-Spiel: wir ziehen 10 Zahlen, berechnen den Mittelwert, wiederholen diesen Vorgang N mal, bekommen N Mittelwerte.

Hier für N = 50:

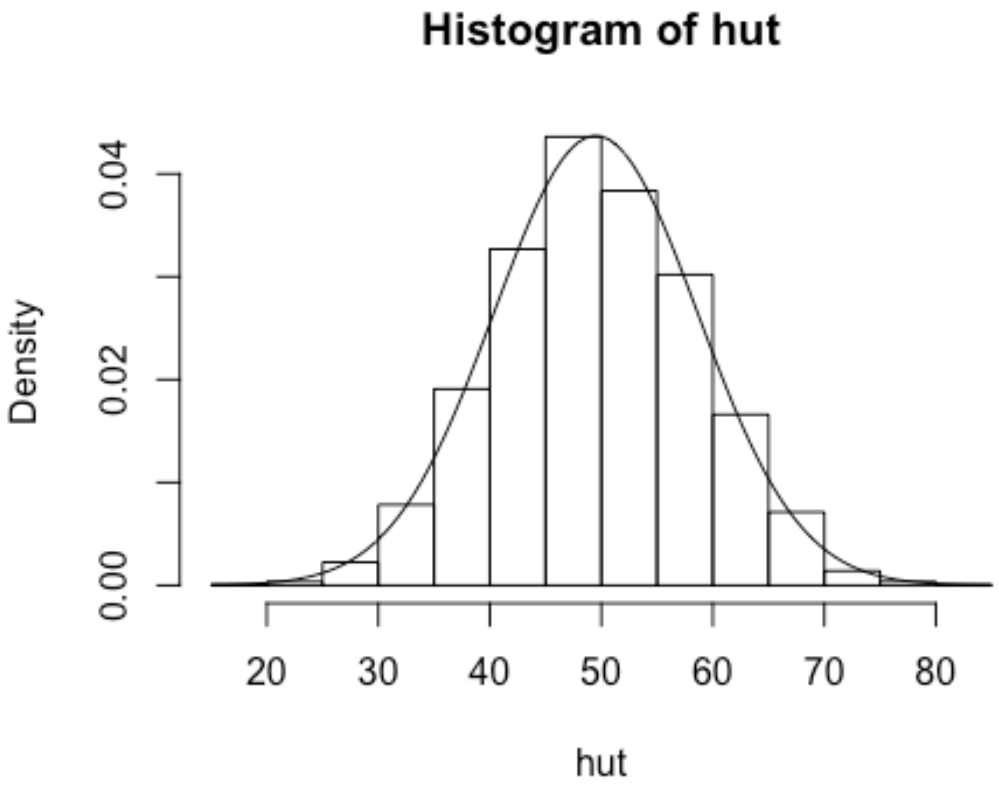
```
hut = proben(0, 99, 10, 50)
mu.hut = mean(0:99)
```

```r
SE.hut = sd(0:99) * sqrt(99/100) / sqrt(10)
hist(hut, freq=F)
# Überlagerung der Normalverteilung
curve(dnorm(x, mu.hut, SE.hut), add=T)
```



Je mehr Stichproben, umso besser die Anpassung an die theoretische Normalverteilung:

```r
# ... hier für N = 5000 Wiederholungen
hut = proben(0, 99, 10, 5000)
mu.hut = mean(0:99)
SE.hut = sd(0:99) * sqrt(99/100) / sqrt(10)
hist(hut, freq=F)
curve(dnorm(x, mu.hut, SE.hut), add=T)
```



8. `pnorm()`: Die proportionale Fläche unter der Normalverteilung

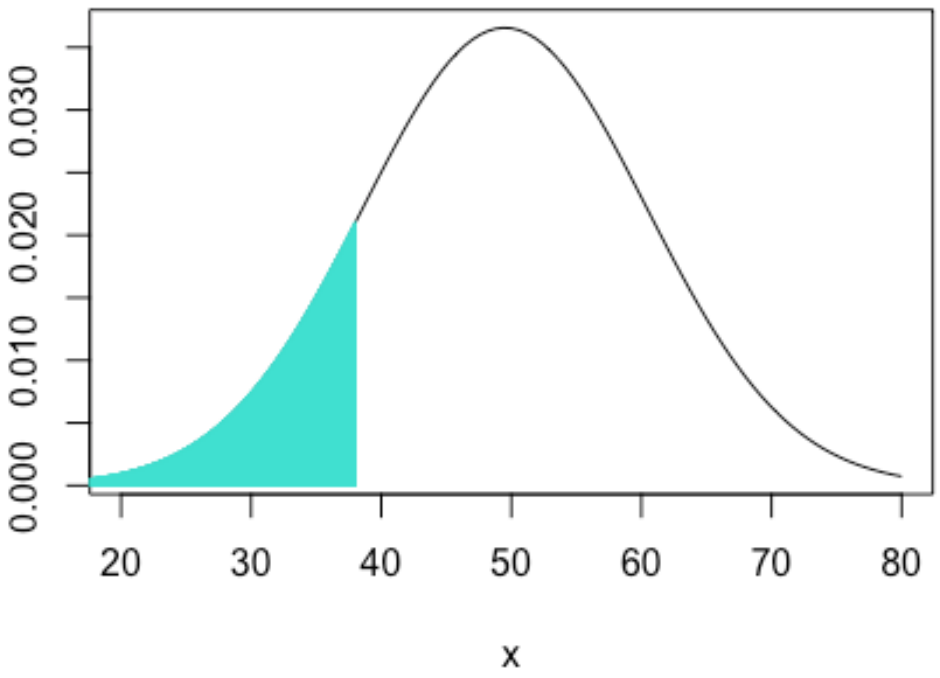
`pnorm()` wird benötigt, um Wahrscheinlichkeiten aus der Normalverteilung zu berechnen.

Damit lassen sich Fragen beantworten wie die Folgende:

Wenn ich 7 Stück Papier aus einem Hut mit den Zahlen 0-99 ziehe, was ist die Wahrscheinlichkeit, dass der Mittelwert unter 38 liegt?

```
# mu, SE müssen berechnet werden
mu = mean(0:99)
SE = sd(0:99) * sqrt(99/100) / sqrt(7)

# Die Antwort graphisch darstellen
# (a) die entsprechende Normalverteilung
xlim = c(20, 80)
curve(dnorm(x, mu, SE), xlim = xlim, ylab = "")
#
# Unter 38: die Fläche unter der Normalverteilung
# zwischen -∞ (minus unendlich) und 38.
# graphisch (so gut wie es geht!)
#
x = seq(15, 38, length=1000)
```

```
y = dnorm(x, mu, SE)
lines(x, y, type="h", col="turquoise")
```

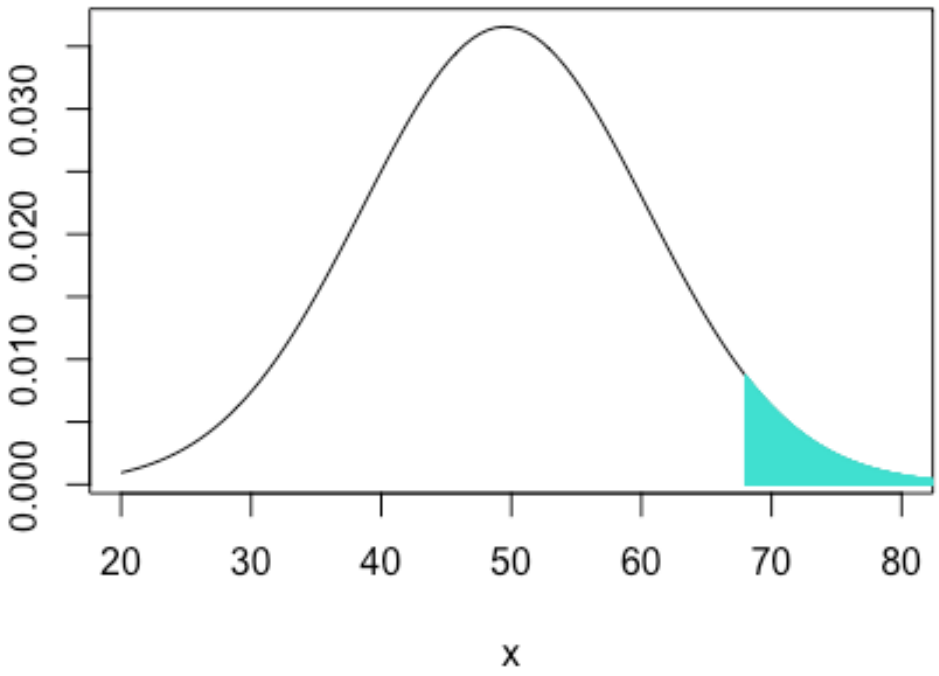



```
pnorm(38, mu, SE)

## [1] 0.1459311

# [1] 0.1459311
```

Das bedeutet: Wenn wir 7 Stück Papier aus einem Hut mit den Zahlen 0, 1, ...99 ziehen, und davon den Mittelwert berechnen, ist in ca. 14-15% der Fälle der Mittelwert unter 38.

Die Wahrscheinlichkeit, dass der Mittelwert über 68 liegt, berechnet sich folgendermaßen:

```
curve(dnorm(x, mu, SE), xlim = xlim, ylab = "")
x = seq(68, 100, length=1000)
y = dnorm(x, mu, SE)
lines(x, y, type="h", col="turquoise")
```



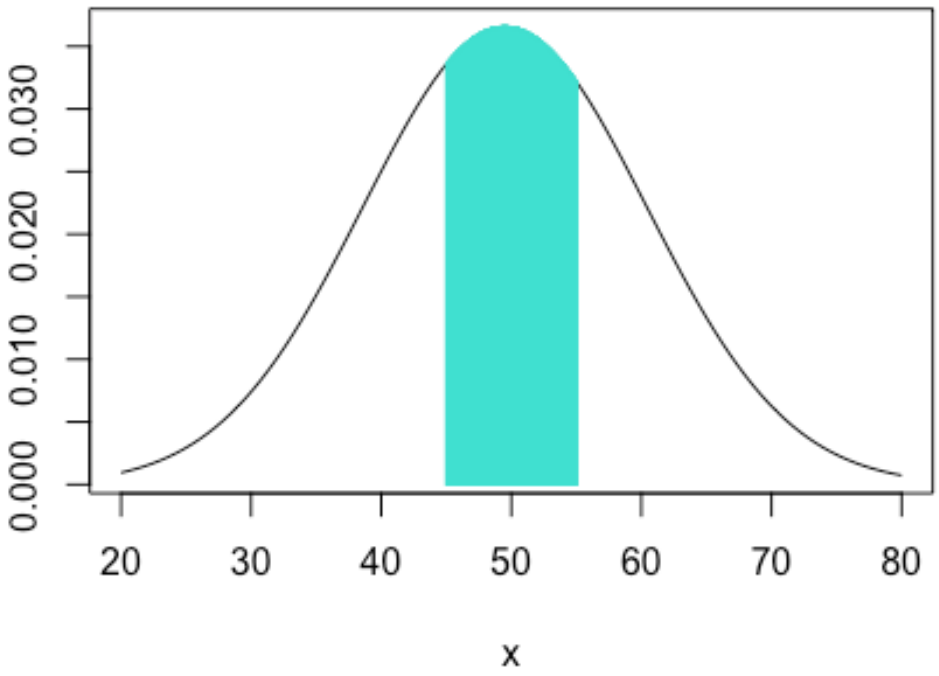
Da die Fläche unter der Normalverteilung 1 (eins) ist, und da `pnorm()` immer die Fläche zwischen -∞ und einen Wert berechnet:

```
1 - pnorm(68, mu, SE)
```

```
## [1] 0.04497725
```

Zwischen 45 und 55?:

```
curve(dnorm(x, mu, SE), xlim = xlim, ylab = "")
x = seq(45, 55, length=1000)
y = dnorm(x, mu, SE)
lines(x, y, type="h", col="turquoise")
```




```
pnorm(55, mu, SE) - pnorm(45, mu, SE)
```

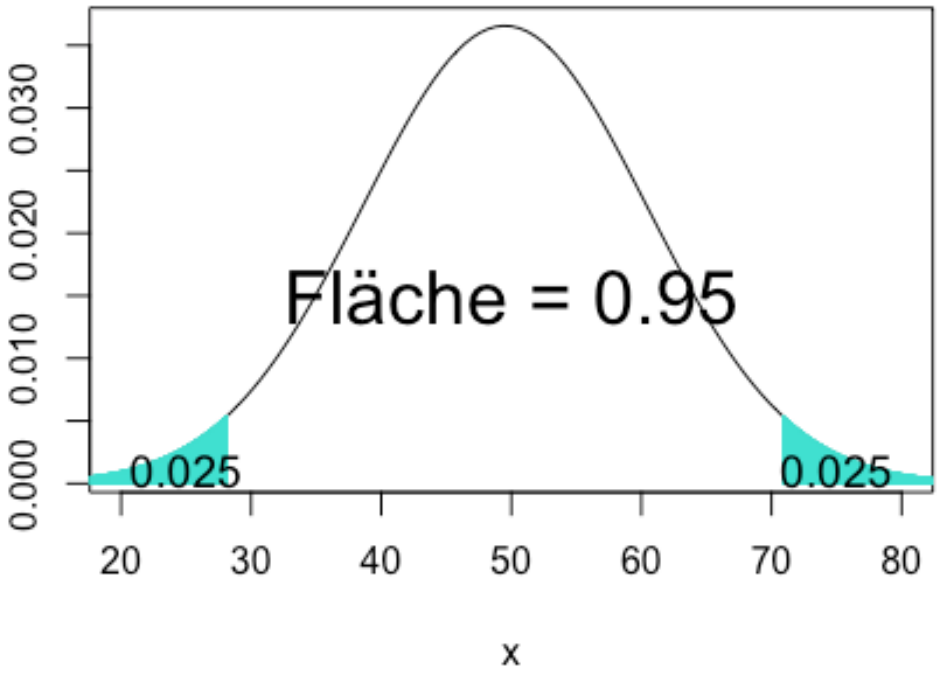
```
## [1] 0.3529035
```

9. `qnorm()` und ein Konfidenzintervall

Ein Hut mit den Zahlen 0, 1, 2, ... 99, aus dem wir 7 Zahlen ziehen und den Mittelwert berechnen: In welchem Bereich liegt dieser Mittelwert (was ist der unterste, was ist der oberste Wert) mit einer Wahrscheinlichkeit von z.B. 95%?

```
curve(dnorm(x, mu, SE), xlim = xlim, ylab = "")

# Einen Wert berechnen, sodass die Fläche zwischen  -∞ und a gleicht 0.025
a = qnorm(0.025, mu, SE)
# Noch einen Wert berechnen, sodass die Fläche zwischen b und  +∞ und a
gleicht 0.025
b = qnorm(0.975, mu, SE)

# graphisch
x = seq(10, a, length=1000)
y = dnorm(x, mu, SE)
curve(dnorm(x, mu, SE), xlim = xlim, ylab = "")
lines(x, y, type="h", col="turquoise")
```

```r
x = seq(b, 90, length=1000)
y = dnorm(x, mu, SE)
lines(x, y, type="h", col="turquoise")
text(25,  0.001, "0.025", cex=1.2)
text(75,  0.001, "0.025", cex=1.2)
text(50, .015, "Fläche = 0.95", cex=2)
```




```r
# d.h.
a
```

```
## [1] 28.11611
```

```r
#[1] 28.11611
b
```

```
## [1] 70.88389
```

```r
#[1] 70.88389
```

Wir ziehen 7 Stück Papier aus einem Hut mit den Zahlen 0-99, und berechnen den Mittelwert. Der Mittelwert liegt zwischen a (28.1) und b (70.9) mit einer Wahrscheinlichkeit von 0.95 (95%).

Oder: das 95%-Konfidenzintervall für den Mittelwert, m, ist: 28.11611 ≤ m ≤ 70.88389 (lies: Der Stichprobenmittelwert ist größer als oder gleich 28.11611 und ist weniger als oder gleich 70.88389 mit einer Wahrscheinlichkeit von 95%).

In der Tat wird in der Regel das 95%-Konfidenzintervall benutzt, um zu bestimmen, ob ein bestimmter Wert zu einer gegebenen Verteilung “dazugehört” oder nicht.