QM6:

Schätzgenauigkeit:

Punktschätzer, Standardfehler (SE) und Konfidenzintervall

Die Differenz als Schätzfehler

je größer die Stichprobe, desto näher sind die Stichprobenmittelwerte am wahren Populationsmittelwert

```
> favstats(flights_clean$arr_delay) # Population
 min  Q1 median Q3  max     mean       sd      n missing
 -86 -17     -5 14 1272 6.895377 44.63329 327346       0
```

<- normalerweise unbekannt

```
     min     Q1 median    Q3 max  mean       sd  n missing .row
...1 -41 -13.00   -1.5 28.25 146  22.3 60.26617 10       0    1
...2 -28  -6.75   16.5 43.75  87  21.5 36.05936 10       0    1
...3 -26  -7.75   10.5 33.25 123  19.3 42.52594 10       0    1
...4 -36 -20.00  -13.5  6.50  14 -10.5 18.00154 10       0    1
...5 -29 -14.25   -5.0  4.75 205  13.3 68.33911 10       0    1
...6 -35 -20.50   -8.0  7.00  83   3.3 37.13055 10       0    1
```

```
     min  Q1 median Q3  max    mean       sd      n missing
...1 -75 -17     -5 14  915 7.04148 44.61456 100000       0
...2 -86 -17     -5 14 1109 6.97302 44.68221 100000       0
...3 -86 -17     -5 14 1272 6.96174 44.85185 100000       0
...4 -74 -17     -5 14 1109 6.91156 44.84390 100000       0
...5 -86 -17     -5 14 1272 6.79285 44.78490 100000       0
...6 -86 -17     -5 14 1127 7.06829 44.74456 100000       0
```

Beobachtung:

- Kennzahlen ‚in der Mitte‘ unserer Stichprobe sind bei kleineren Stichproben bessere Schätzer (Median, iqr)
- Aggregierende Kennzahlen haben die Tendenz zur Mitte, der MW (mean) ist deshalb der häufigste Schätzer

Fig.: Verteilung der Differenz zwischen population_mean und Stichprobenmittelwerte

Die Differenz als Schätzfehler

je größer die Stichprobe, desto näher sind die Stichprobenmittelwerte am wahren Populationsmittelwert

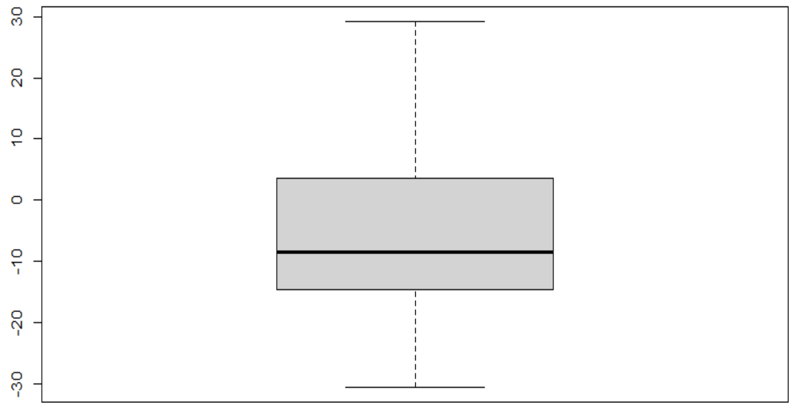
```
> favstats(flights_clean$arr_delay) # Population
 min  Q1 median Q3  max     mean       sd      n missing
 -86 -17     -5 14 1272 6.895377 44.63329 327346       0
```

<- normalerweise unbekannt

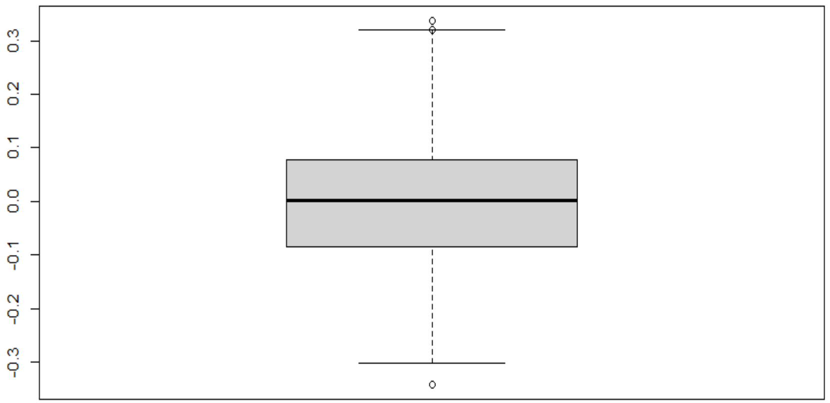
	min	Q1	median	Q3	max	mean	sd	n	missing	.row
...1	-41	-13.00	-1.5	28.25	146	22.3	60.26617	10	0	1
...2	-28	-6.75	16.5	43.75	87	21.5	36.05936	10	0	1
...3	-26	-7.75	10.5	33.25	123	19.3	42.52594	10	0	1
...4	-36	-20.00	-13.5	6.50	14	-10.5	18.00154	10	0	1
...5	-29	-14.25	-5.0	4.75	205	13.3	68.33911	10	0	1
...6	-35	-20.50	-8.0	7.00	83	3.3	37.13055	10	0	1

	min	Q1	median	Q3	max	mean	sd	n	missing
...1	-75	-17	-5	14	915	7.04148	44.61456	100000	0
...2	-86	-17	-5	14	1109	6.97302	44.68221	100000	0
...3	-86	-17	-5	14	1272	6.96174	44.85185	100000	0
...4	-74	-17	-5	14	1109	6.91156	44.84390	100000	0
...5	-86	-17	-5	14	1272	6.79285	44.78490	100000	0
...6	-86	-17	-5	14	1127	7.06829	44.74456	100000	0

starke Streuung (Bsp.)



uns interessiert das 95%- Intervall der Schwankung des means

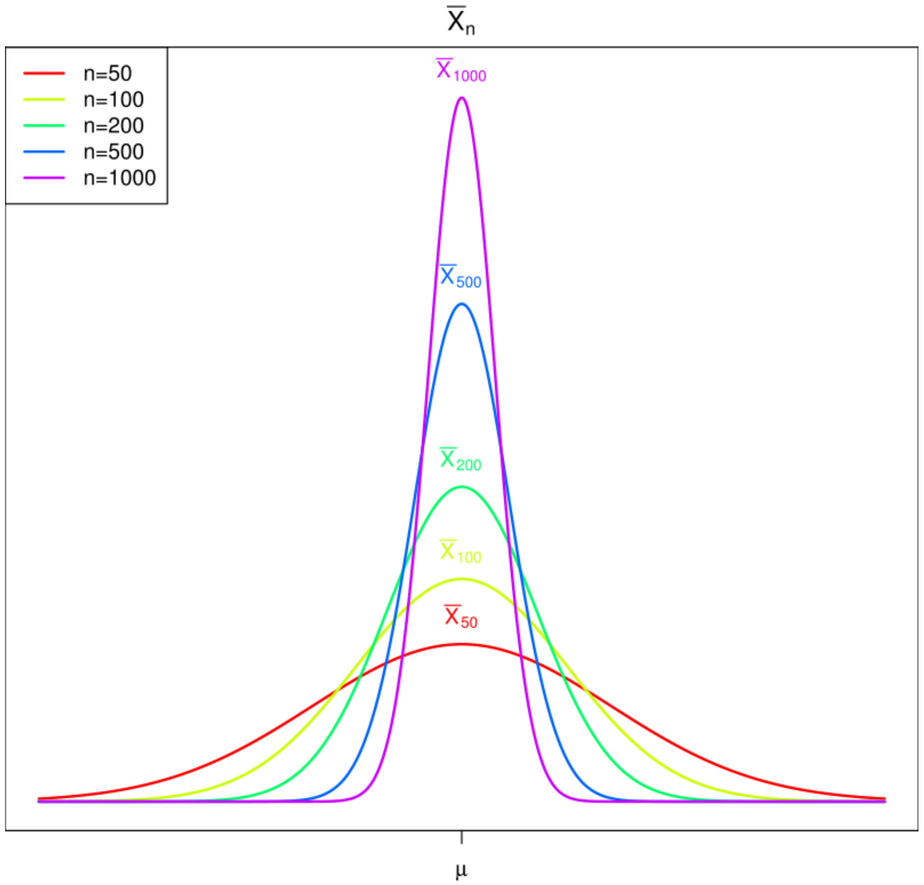


weniger Streuung

Fig.: Verteilung der Differenz zwischen population_mean und Stichprobenmittelwerte

Verteilung von Stichprobenmittelwerten

z.B. ‚flights'



μ = 6.98 min
(z.B. wie in flights)

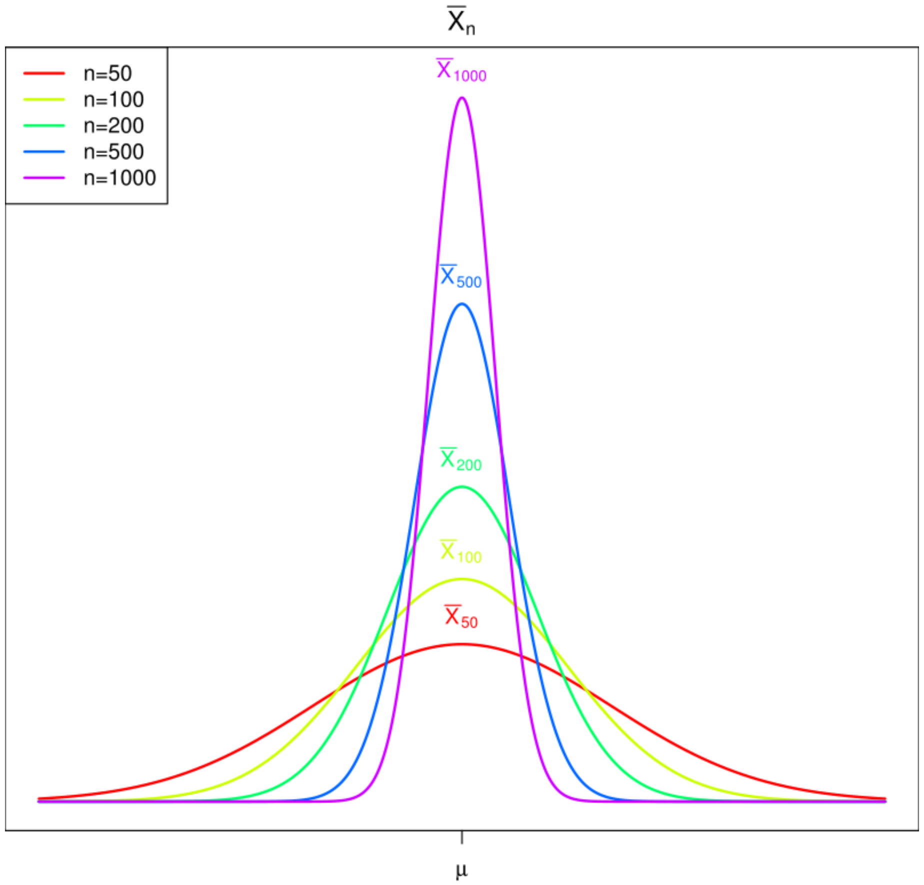
- Wir können nie wissen wie nah unser MW aus dem sample am wahren μ liegt, da μ in der Regel gesucht wird.

- Je größer die Stichprobe, desto schmaler ist die Verteilung der Stichprobenmittelwerte. Dadurch wird der Populationsmittelwert präziser geschätzt. Die Breite der Stichprobenverteilung wird durch die Streuung/ sd bestimmt.

Quelle: https://de.wikipedia.org/wiki/Erwartungstreue#/media/Datei:
Distribution_of_mean_estimator.svg

Verteilung von Stichprobenmittelwerten

z.B. ‚flights‘



μ = 6.98 min
(z.B. wie in flights)

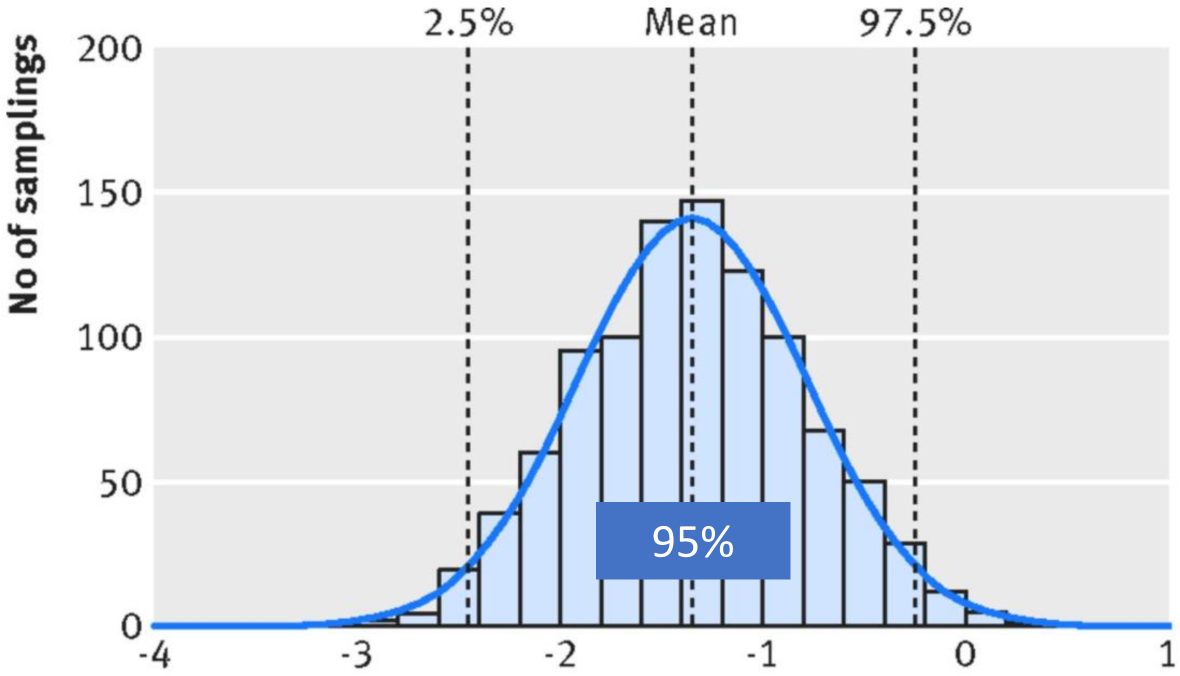
- **Wir benötigen demnach nur die sd der Stichprobenmittelwerte um die Genauigkeit unserer Punktschätzung anzugeben**

- Vorgehen:

1) Standardfehler berechnen (theoretisch oder praktisch)
2) 95%- Intervall berechnen

Quelle: https://de.wikipedia.org/wiki/Erwartungstreue#/media/Datei:
Distribution_of_mean_estimator.svg

Punktschätzer und Konfidenzintervall

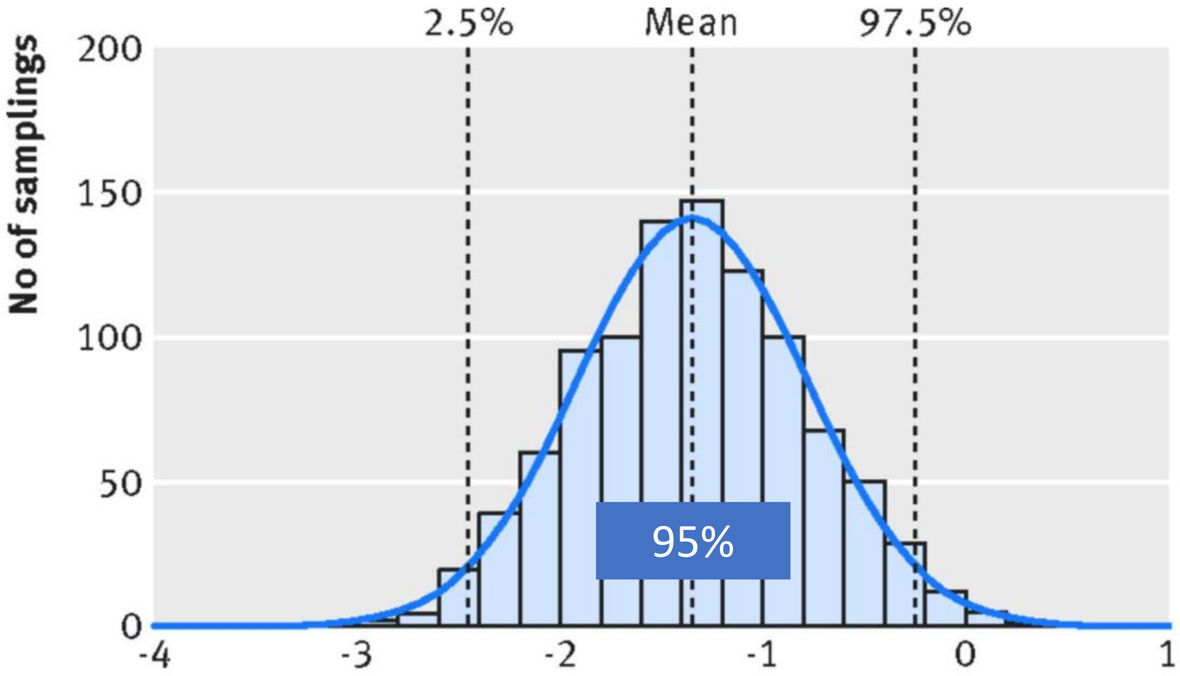


- Die Schätzgenauigkeit kann man oft besser durch eine untere und eine obere Grenze einordnen.

- Der Stichprobenmittelwert gilt dabei als **Punktschätzer.**
- Das Intervall nennt sich Vertrauensintervall, oder häufiger: **Konfidenzintervall**.

Figure: https://www.bmj.com/content/350/bmj.h2622

Punktschätzer und Konfidenzintervall



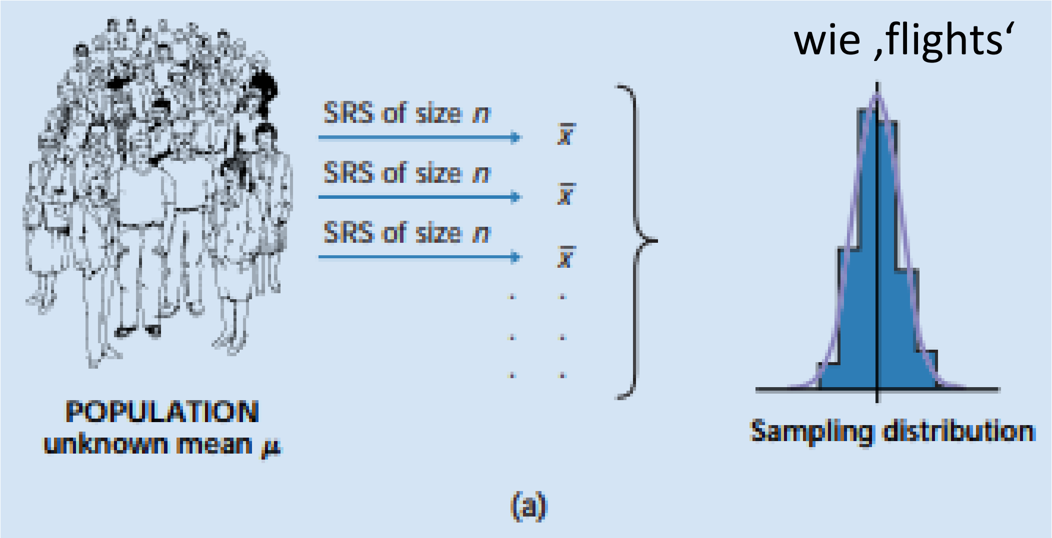
- Die Schätzgenauigkeit kann man oft besser durch eine untere und eine obere Grenze einordnen.

- Der Stichprobenmittelwert gilt dabei als **Punktschätzer.**
- Das Intervall nennt sich Vertrauensintervall, oder häufiger: **Konfidenzintervall**.

$$x_u = \bar{x} - z \cdot \frac{\sigma}{\sqrt{n}} \qquad x_o = \bar{x} + z \cdot \frac{\sigma}{\sqrt{n}}$$

Mittels des Standardfehlers SE wird häufig die 95%-Umgebung um den Punktschätzer angegeben.

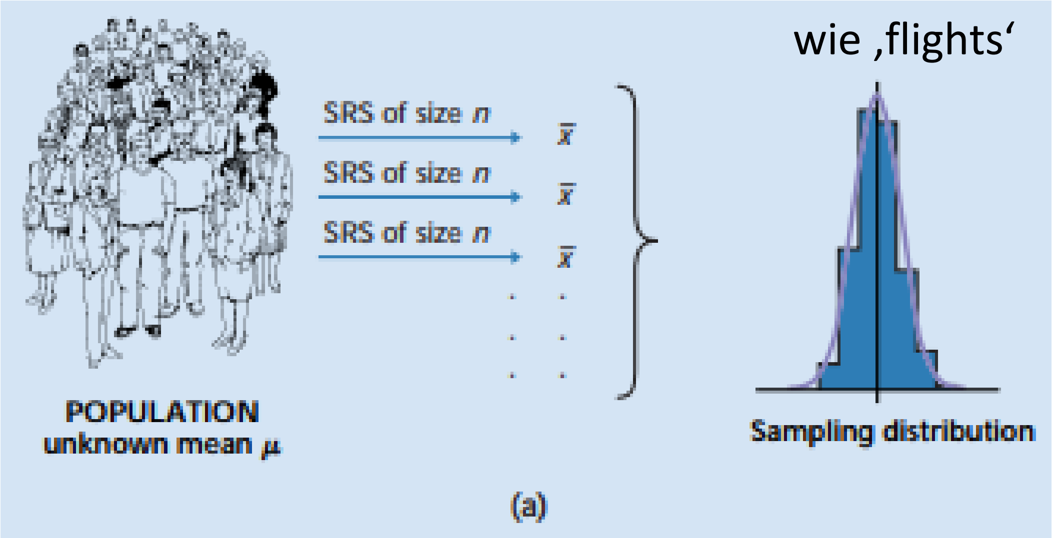
Herangehensweise: empirische oder theoretische **Berechnung des Standardfehlers** (standard error: SE)



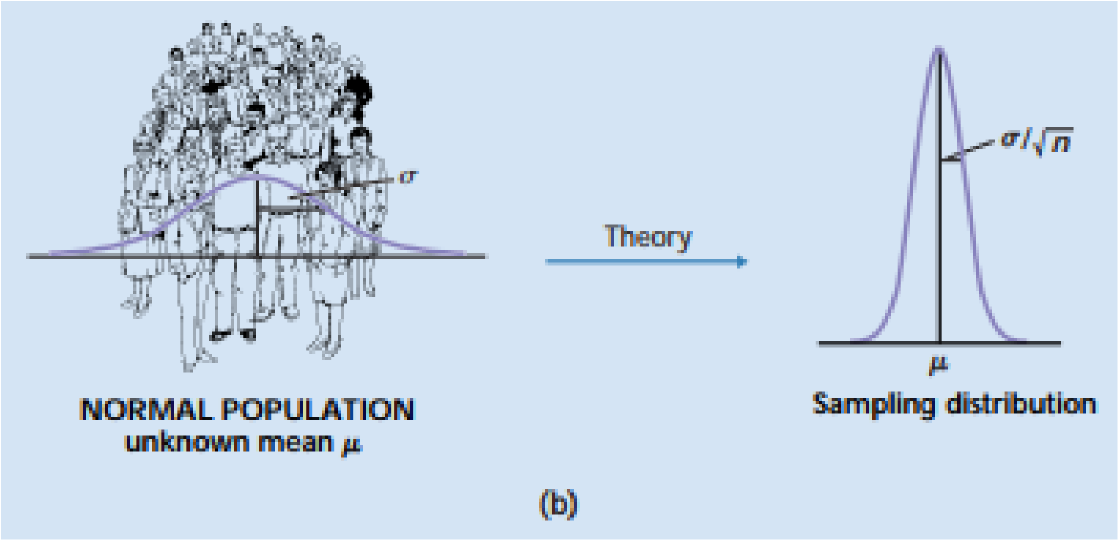
Praktische Berechnung:
ziehe samples
berechne means
berechne sd dieser means (=Standardfehler)

Quelle: http://statweb.stanford.edu/~tibs/stat315a/Supplements/bootstrap.pdf

Herangehensweise: empirische oder theoretische **Berechnung des Standardfehlers** (standard error: SE)



Praktische Berechnung:
ziehe samples
berechne means
berechne sd dieser means (=Standardfehler)



Theoretische Berechnung:
ziehe ein sample
berechne sd
teile sd durch Wurzel aus n (=Standardfehler)

$$\boxed{SE_{\bar{x}} = \frac{s}{\sqrt{n}}}$$

Quelle: http://statweb.stanford.edu/~tibs/stat315a/Supplements/bootstrap.pdf

Kleine(re) Stichprobe (n=100)

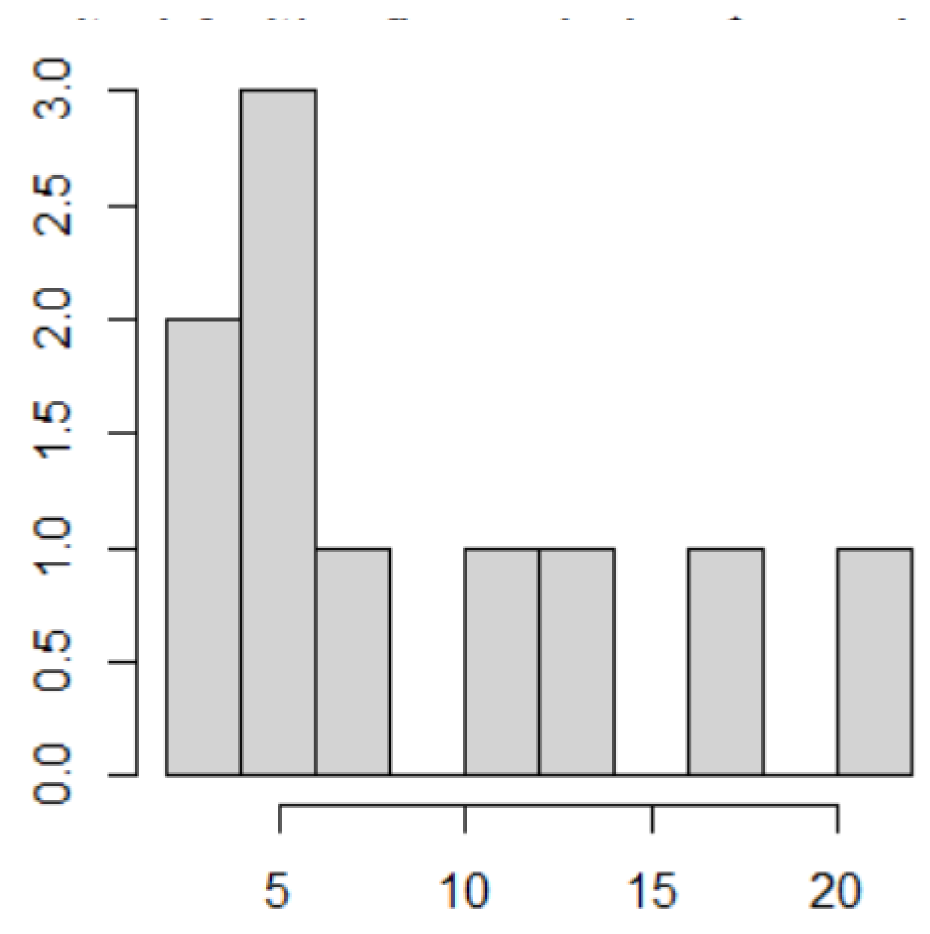
Console Terminal Jobs

~/P2/

```
> # erzeuge x samples der Große size und zeige favstats
> fl_sample_loop <- do(10) *
+   favstats(sample(flights_clean$arr_delay, size = 100))
>
> hist(fl_sample_loop$mean, breaks=10) #Histogramm
> plot(density(fl_sample_loop$mean,bw=1)) #Glättung Dichteplot
>
> mu<-mean(flights_clean$arr_delay);mu # Mittelwert der population
[1] 6.895377
> abline(v = mean(flights_clean$arr_delay), col = 4) #mean population
>
> m_<-mean(fl_sample_loop$mean);m_ # MW der Mittelwerte der samples
[1] 9.044
> abline(v = mean(fl_sample_loop$mean), col = 2) #mean StichprobenMW
>
> ## 2.5% und 97.5% quantile (mittleren 95%)
> q<-quantile(fl_sample_loop$mean, prob=c(0.025,0.975));q
    2.5%    97.5%
 3.79700 19.64875
> abline(v = q, col = 2, lty = 2)
>
> sd(fl_sample_loop$mean) #Standardfehler, empirisch
[1] 5.991919
>
> # Berechnung des theoretischen SE = sd/Wurzel n
> se_t<-sd(sample(flights_clean$arr_delay, size = 100))/sqrt(100);se_t
[1] 3.106828
```

Files Plots Packages Help Viewer

Zoom Export



Kleine(re) Stichprobe (n=100)

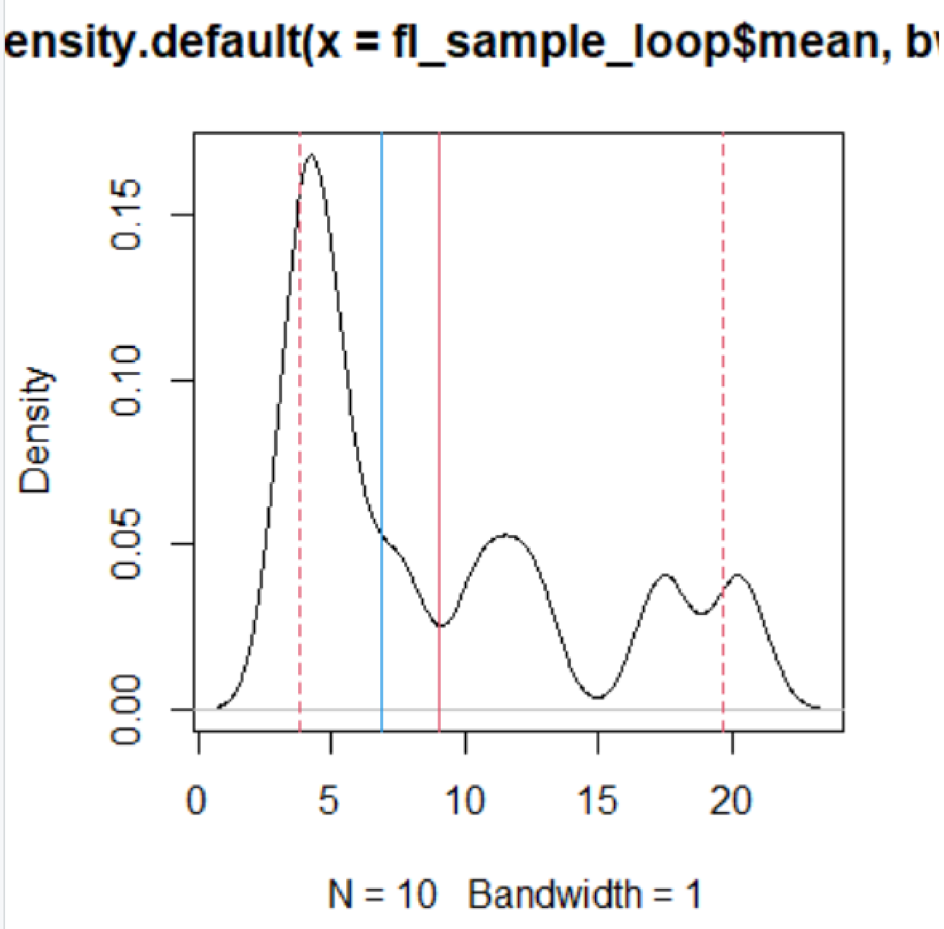
Console Terminal Jobs

~/P2/

```
> # erzeuge x samples der Große size und zeige favstats
> fl_sample_loop <- do(10) *
+   favstats(sample(flights_clean$arr_delay, size = 100))
>
> hist(fl_sample_loop$mean, breaks=10) #Histogramm
> plot(density(fl_sample_loop$mean,bw=1)) #Glättung Dichteplot
>
> mu<-mean(flights_clean$arr_delay);mu # Mittelwert der population
[1] 6.895377
> abline(v = mean(flights_clean$arr_delay), col = 4) #mean population
>
> m_<-mean(fl_sample_loop$mean);m_ # MW der Mittelwerte der samples
[1] 9.044
> abline(v = mean(fl_sample_loop$mean), col = 2) #mean StichprobenMW
>
> ## 2.5% und 97.5% quantile (mittleren 95%)
> q<-quantile(fl_sample_loop$mean, prob=c(0.025,0.975));q
    2.5%    97.5%
 3.79700 19.64875
> abline(v = q, col = 2, lty = 2)
>
> sd(fl_sample_loop$mean) #Standardfehler, empirisch
[1] 5.991919
>
> # Berechnung des theoretischen SE = sd/Wurzel n
> se_t<-sd(sample(flights_clean$arr_delay, size = 100))/sqrt(100);se_t
[1] 3.106828
```

Files Plots Packages Help Viewer

Zoom Export

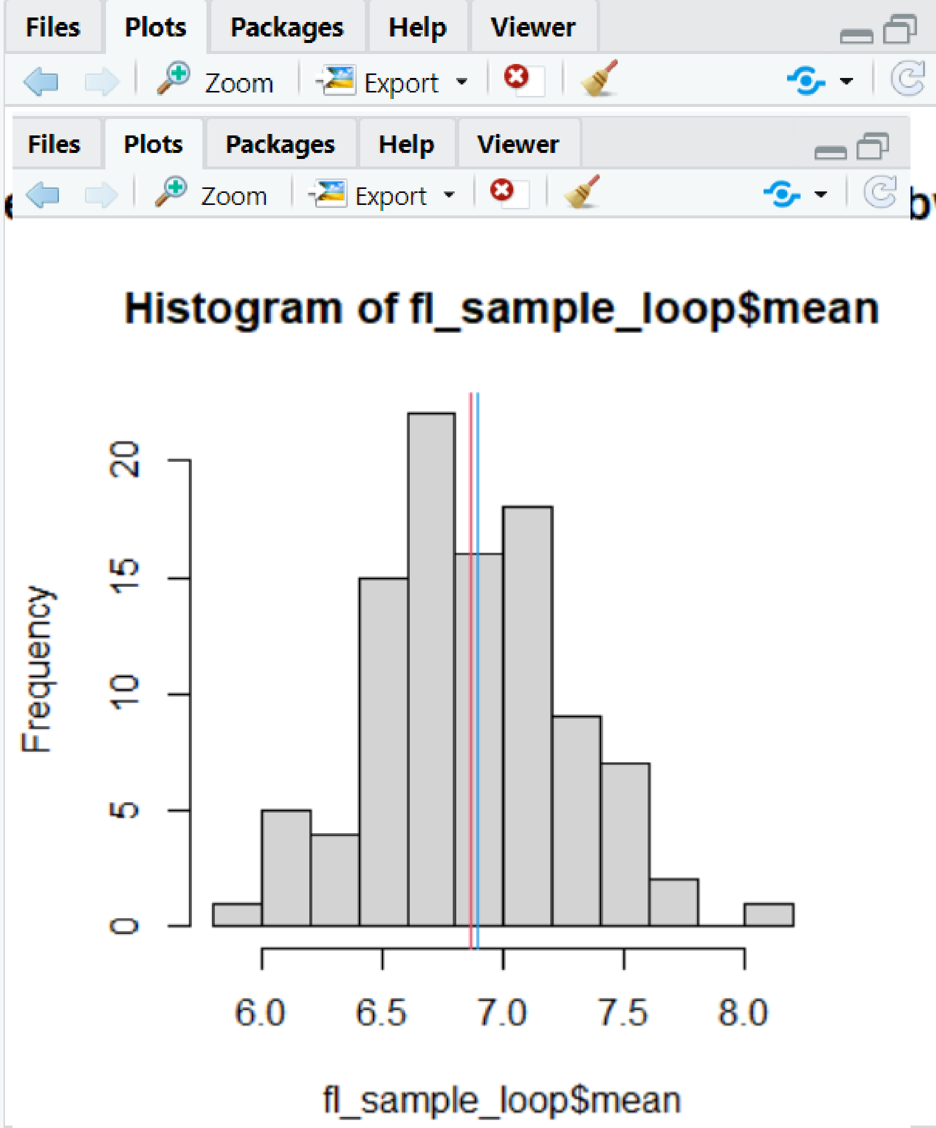


Große Stichprobe (n=10.000)

Console Terminal Jobs

~/P2/

```
> # erzeuge x samples der Große size und zeige favstats
> fl_sample_loop <- do(100) *
+   favstats(sample(flights_clean$arr_delay, size = 10000))
>
> hist(fl_sample_loop$mean, breaks=10) #Histogramm
> plot(density(fl_sample_loop$mean,bw=1)) #Glättung Dichteplot
>
> mu<-mean(flights_clean$arr_delay);mu # Mittelwert der population
[1] 6.895377
> abline(v = mean(flights_clean$arr_delay), col = 4) #mean population
>
> m_<-mean(fl_sample_loop$mean);m_ # MW der Mittelwerte der samples
[1] 6.967949
> abline(v = mean(fl_sample_loop$mean), col = 2) #mean StichprobenMW
>
> ## 2.5% und 97.5% quantile (mittleren 95%)
> q<-quantile(fl_sample_loop$mean, prob=c(0.025,0.975));q
    2.5%    97.5%
5.965695 7.818585
> abline(v = q, col = 2, lty = 2)
>
> sd(fl_sample_loop$mean) #Standardfehler, empirisch
[1] 0.4835039
>
> # Berechnung des theoretischen SE = sd/Wurzel n
> se_t<-sd(sample(flights_clean$arr_delay, size = 10000))/sqrt(10000);se_t
[1] 0.4513942
```



Große Stichprobe (n=10.000)

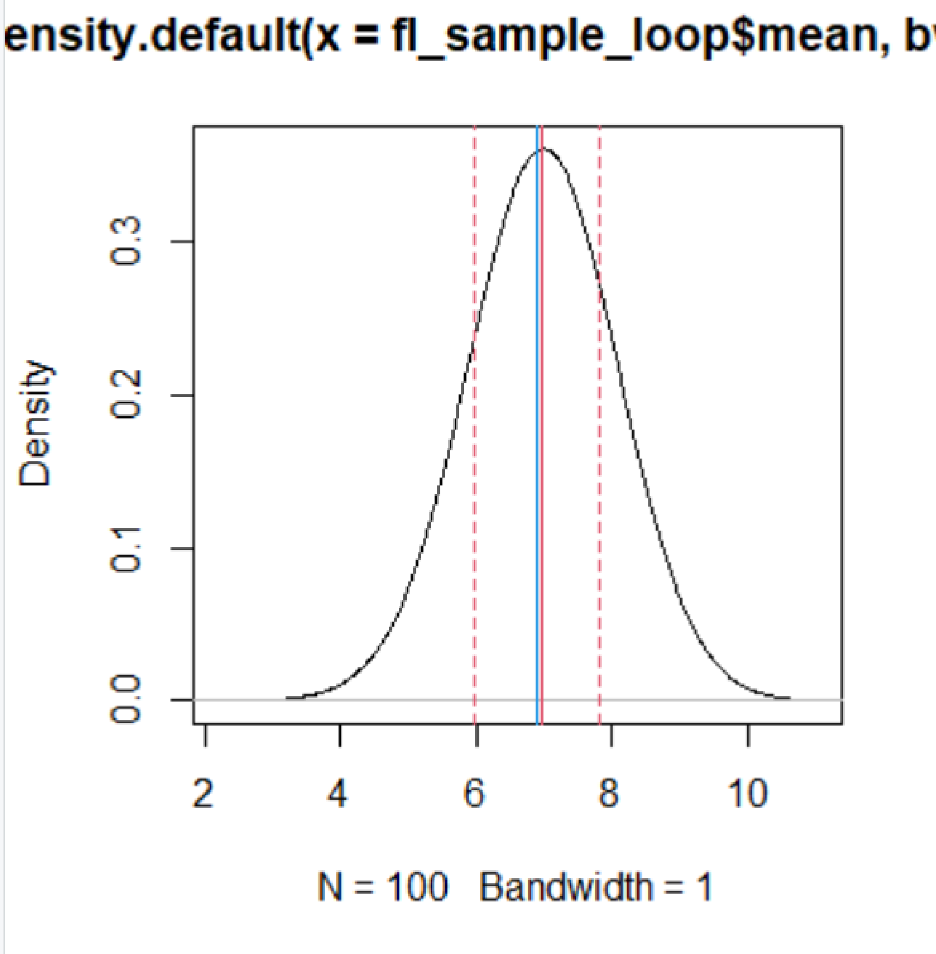
Console | Terminal | Jobs

~/P2/

```
> # erzeuge x samples der Große size und zeige favstats
> fl_sample_loop <- do(100) *
+   favstats(sample(flights_clean$arr_delay, size = 10000))
>
> hist(fl_sample_loop$mean, breaks=10) #Histogramm
> plot(density(fl_sample_loop$mean,bw=1)) #Glättung Dichteplot
>
> mu<-mean(flights_clean$arr_delay);mu # Mittelwert der population
[1] 6.895377
> abline(v = mean(flights_clean$arr_delay), col = 4) #mean population
>
> m_<-mean(fl_sample_loop$mean);m_ # MW der Mittelwerte der samples
[1] 6.967949
> abline(v = mean(fl_sample_loop$mean), col = 2) #mean StichprobenMW
>
> ## 2.5% und 97.5% quantile (mittleren 95%)
> q<-quantile(fl_sample_loop$mean, prob=c(0.025,0.975));q
    2.5%    97.5%
5.965695 7.818585
> abline(v = q, col = 2, lty = 2)
>
> sd(fl_sample_loop$mean) #Standardfehler, empirisch
[1] 0.4835039
>
> # Berechnung des theoretischen SE = sd/Wurzel n
> se_t<-sd(sample(flights_clean$arr_delay, size = 10000))/sqrt(10000);se_t
[1] 0.4513942
```

Files | Plots | Packages | Help | Viewer

Zoom | Export



QM7

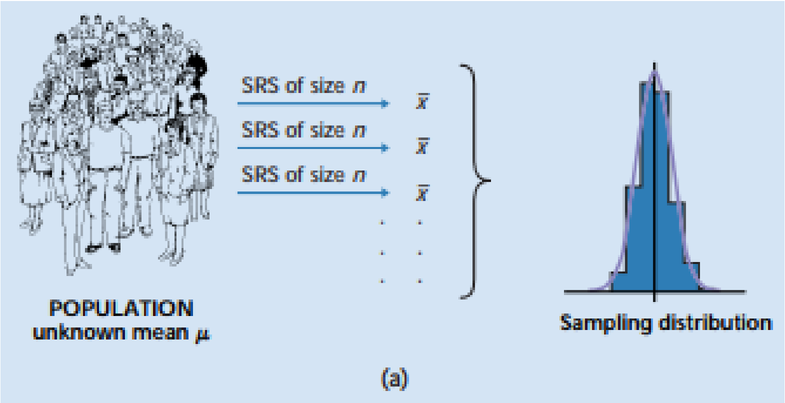
Einführung simulationsbasierte Inferenz (SBI):

Bootstrap (resample) Konfidenzintervall

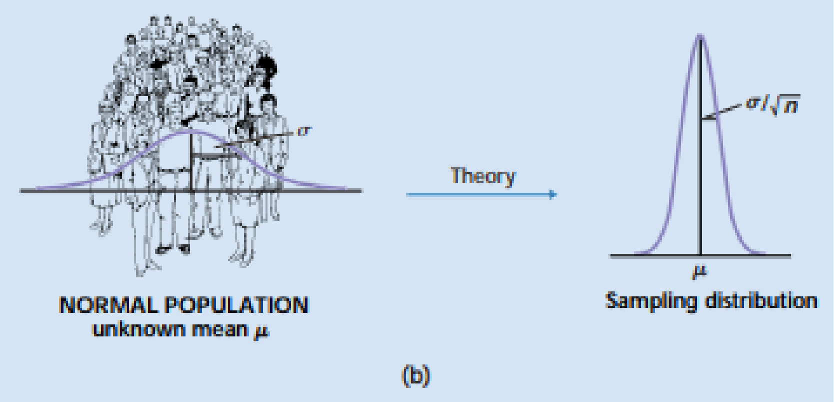
Schätzen vs. Hypothesen testen

Ein Ausreißertest

Herangehensweise: empirische oder theoretische **Berechnung des Standardfehlers** (standard error: SE)

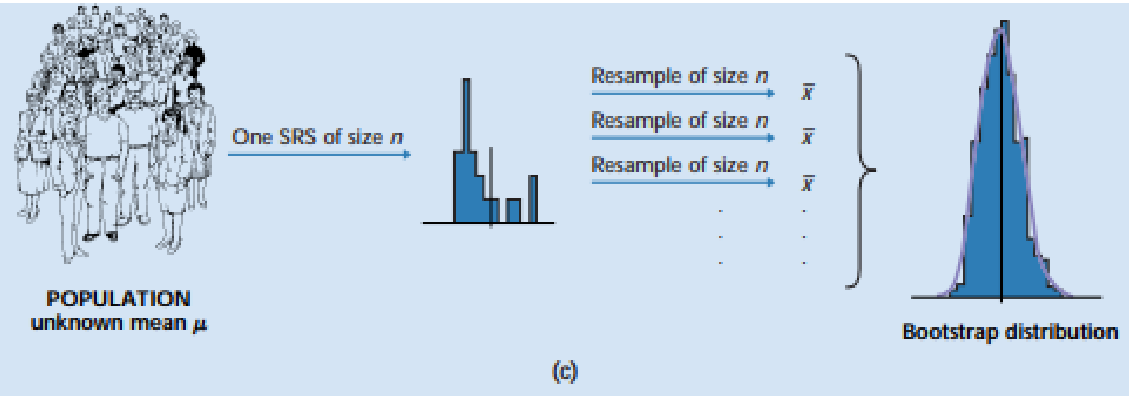


Praktische Berechnung:
ziehe samples
berechne means
berechne sd dieser means (=Standardfehler)



Theoretische Berechnung:
ziehe ein sample
berechne sd
teile sd durch Wurzel aus n (=Standardfehler)

$$\boxed{SE_{\bar{x}} = \frac{s}{\sqrt{n}}}$$



<- QM7: Resampling

Quelle: http://statweb.stanford.edu/~tibs/stat315a/Supplements/bootstrap.pdf

samplen vs. resamplen

QM5c.R | Funktionen.R | QM6c QM7.R*

Source on Save | Run | Source

```r
### Resamplen: mehrfaches Ziehen aus EINER Stichprobe #######++++++++++

# Schritt 1: ziehe EIN sample

fl_sample<- sample(flights_clean$arr_delay, size = 3) #2500
fl_sample
mean(fl_sample)

# Schritt 2: resample: Ziehen mit Zurücklegen aus unserem (einzigen) sampl
fl_resample <- resample(fl_sample, size = 3)
fl_resample
```

19:1 (Top Level) R Script

Console | Terminal | Jobs

~/P2/

```
> fl_sample<- sample(flights_clean$arr_delay, size = 3) #2500
> fl_sample
[1] -12 -29 128
> fl_resample <- resample(fl_sample, size = 3)
> fl_resample
[1] 128 -29 128
>
```

Mehrfache können gezogen werden

Ziehe ein sample der Größe = 3

Ziehe daraus ein sample der Größe 3. Zwischendurch zurücklegen

Ziehen ohne Zurücklegen (samplen)

Ziehen mit Zurücklegen (resamplen)

Wiederholtes resamplen (bootstrap)

```r
##################################################
# als loop
# etwas Größeres sample ziehen
fl_sample<- sample(flights_clean$arr_delay, size = 2500)
mean(fl_sample) #an dieser Stelle läßt sich nur der mean des samples berechnen

#aus diesem sample resamplen um die Breite des Konf-intervalls zu schätzen
set.seed(2021)

fl_resample_loop <- do (100000)*mean(resample(fl_sample)) #for-loop
mean(fl_resample_loop$mean) #mean der means (grand mean)
sd(fl_resample_loop$mean) #Standardfehler
```

```
> mean(fl_sample)
[1] 6.8636
> set.seed(2021)
>
> fl_resample_loop <- do (100000)*mean(resample(fl_sample))
> mean(fl_resample_loop$mean)
[1] 6.862183
> sd(fl_resample_loop$mean)
[1] 0.9409522
>
```

Wiederholtes resamplen (bootstrap)

```
hist(fl_resample_loop$mean, breaks=50) #Histogramm
plot(density(fl_resample_loop$mean,bw=1)) #Glättung Dichteplot

mu<-mean(flights_clean$arr_delay);mu # Mittelwert der population
abline(v = mean(flights_clean$arr_delay), col = 4) #mean population

m_<-mean(fl_resample_loop$mean);m_ # MW der Mittelwerte der samples
abline(v = mean(fl_resample_loop$mean), col = 2) #mean StichprobenMW

## 2.5% und 97.5% quantile (mittleren 95%)
q<-quantile(fl_resample_loop$mean, prob=c(0.025,0.975));q
abline(v = q, col = 2, lty = 2)

paste("Der Punktschätzer beträgt",mean(fl_resample_loop$mean),". Der wahre unbek
      in 95% der Stichproben zwischen ",q[1]," und",q[2])
```

37:1 (Untitled) R Script

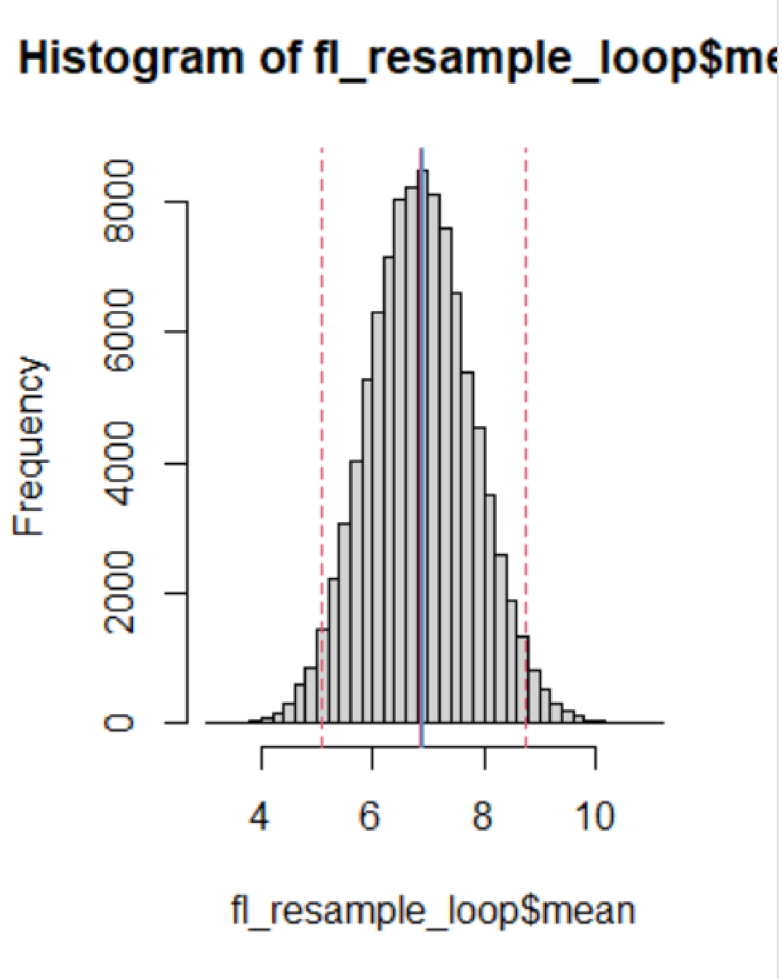
Console Terminal Jobs

~/P2/

```
> mu<-mean(flights_clean$arr_delay);mu # Mittelwert der population
[1] 6.895377
> m_<-mean(fl_resample_loop$mean);m_ # MW der Mittelwerte der samples
[1] 6.862183
> paste("Der Punktschätzer beträgt",mean(fl_resample_loop$mean),". Der wahre unbekannt
e Populationsparameter µ liegt
+       in 95% der Stichproben zwischen ",q[1]," und",q[2])
[1] "Der Punktschätzer beträgt 6.862183416 . Der wahre unbekannte Populationsparameter
 µ liegt\n       in 95% der Stichproben zwischen  5.06519  und 8.74161"
```

Files Plots Packages Help Viewer

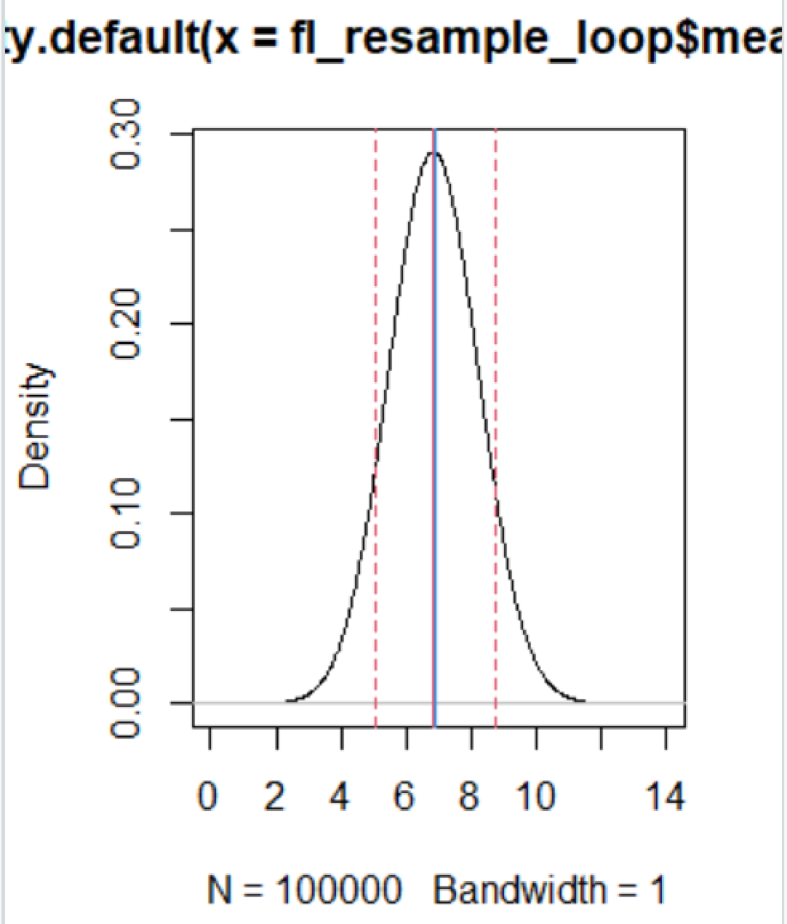
Zoom Export



Wiederholtes resamplen (bootstrap)

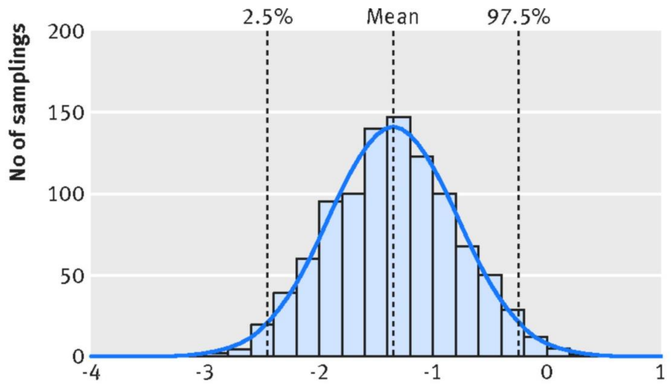
```r
hist(fl_resample_loop$mean, breaks=50) #Histogramm
plot(density(fl_resample_loop$mean,bw=1)) #Glättung Dichteplot

mu<-mean(flights_clean$arr_delay);mu # Mittelwert der population
abline(v = mean(flights_clean$arr_delay), col = 4) #mean population

m_<-mean(fl_resample_loop$mean);m_ # MW der Mittelwerte der samples
abline(v = mean(fl_resample_loop$mean), col = 2) #mean StichprobenMW

## 2.5% und 97.5% quantile (mittleren 95%)
q<-quantile(fl_resample_loop$mean, prob=c(0.025,0.975));q
abline(v = q, col = 2, lty = 2)

paste("Der Punktschätzer beträgt",mean(fl_resample_loop$mean),". Der wahre unbek
      in 95% der Stichproben zwischen ",q[1]," und",q[2])
```

```
> mu<-mean(flights_clean$arr_delay);mu # Mittelwert der population
[1] 6.895377
> m_<-mean(fl_resample_loop$mean);m_ # MW der Mittelwerte der samples
[1] 6.862183
> paste("Der Punktschätzer beträgt",mean(fl_resample_loop$mean),". Der wahre unbekannt
e Populationsparameter µ liegt
+       in 95% der Stichproben zwischen ",q[1]," und",q[2])
[1] "Der Punktschätzer beträgt 6.862183416 . Der wahre unbekannte Populationsparameter
 µ liegt\n       in 95% der Stichproben zwischen  5.06519  und 8.74161"
```



Schätzen vs. Hypothesen prüfen

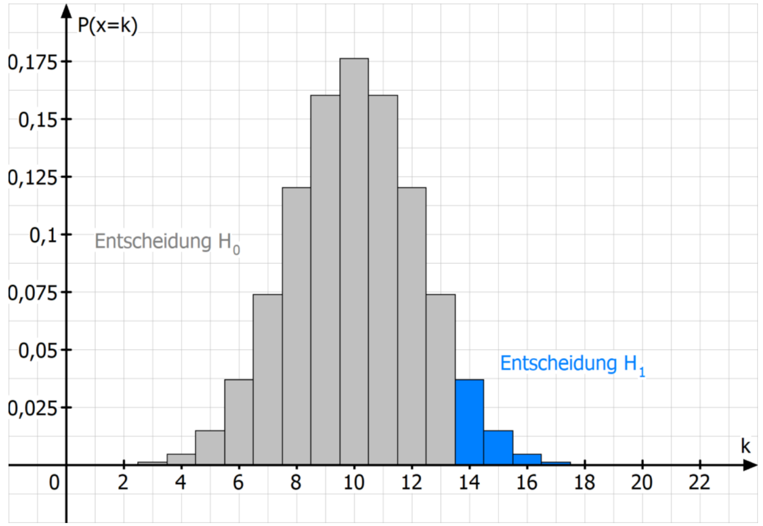
- Estimation (Schätzung)-> Confidence interval
  - In welchem Intervall liegt der unbekannte Wert?



- Decision (Entscheidung) -> Hypothesis test

Forschungsfragen beantworten (z.B.):
Haben die Flüge am JFK eine größere Verspätung?
(angenommen die Gesamtpopulation würde nicht vorliegen)

Hypothesenpaar formulieren
H_1: Die Flüge am JFK haben eine größere Verspätung
H_0: Die Flüge am JFK haben **k**eine größere Verspätung
(in der Nullhypothese wird von Gleichheit ausgegangen)

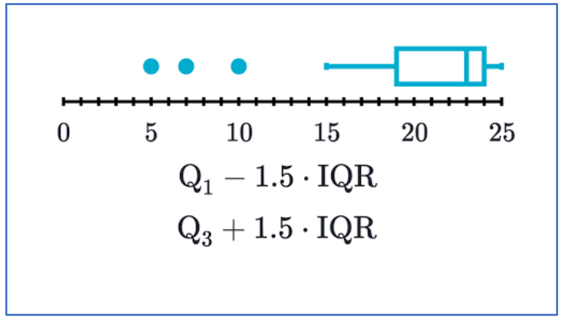


→ QM8

Ein Ausreißertest

Ausreißertest nach IQR

```r
# Ausreißertest: Es liegt eine Messung von Körpergrößen vor

height <- c(1.78,1.69,1.81,1.83,1.80,1.40) #c für Vektor
favstats(height) #Kennzahlen anzeigen
boxplot(height) #visualisieren
# der Wert 1.40 m wird hier als Ausreißer (outlier) gesehen, siehe
# Punkt unten unterhalb des boxplots
#iqr_test (nach boxplot), untere und obere Grenze
quantile(height) # wir brauchen Q1 und Q3

iqr_test_u <- function(x){
  quantile(x)[2]-1.5*(iqr(x))    #Q1-1.5*IQR
}
iqr_test_o <- function(x){
  quantile(x)[4]+1.5*(iqr(x))    #Q3+1.5*IQR
}
u<-iqr_test_u(height);u
o<-iqr_test_o(height);o
```



20:24 (Top Level) R Script

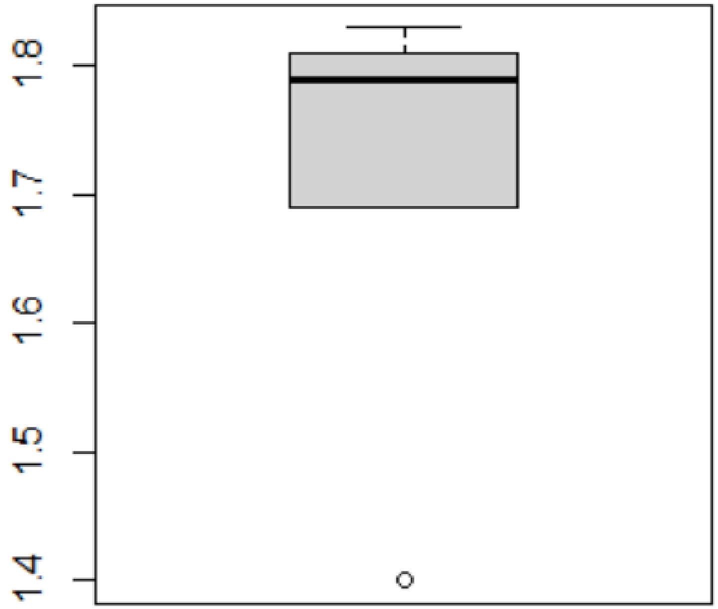
Console Terminal Jobs

~/P2/

```
 min      Q1 median     Q3  max     mean        sd n missing
 1.4 1.7125   1.79 1.8075 1.83 1.718333 0.1633911 6       0
> quantile(height)
    0%     25%     50%     75%    100%
1.4000 1.7125 1.7900 1.8075 1.8300
```

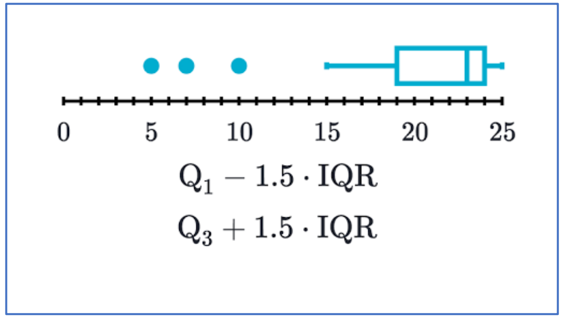
Files Plots Packages Help Viewer

Zoom Export



Ausreißertest nach IQR

```r
#iqr_test (nach boxplot), untere und obere Grenze
quantile(height) # wir brauchen Q1 und Q3

iqr_test_u <- function(x){
  quantile(x)[2]-1.5*(iqr(x))    #Q1-1.5*IQR
}
iqr_test_o <- function(x){
  quantile(x)[4]+1.5*(iqr(x))    #Q3+1.5*IQR
}
u<-iqr_test_u(height);u
o<-iqr_test_o(height);o

boxplot(height,ylim=c(0.9*min(height),1.1*max(height)))
abline(h=u); abline(h=o)
```




```
> u<-iqr_test_u(height);u
 25%
1.57
> o<-iqr_test_o(height);o
 75%
1.95
> boxplot(height,ylim=c(0.9*min(height),1.1*max(height)))
> abline(h=u); abline(h=o)
```